

# SCORM® Users Guide for Instructional Designers

---

**SCORM**  
**2004**  
**4<sup>th</sup> Edition**

Version 8.2

January 3, 2012



**The Power of Global Collaboration**  
Defense | Government | Industry | Academia

# Brief Table of Contents

---

<b>Chapter 1. SCORM in a Nutshell.....</b>	<b>6</b>
<b>Chapter 2. Overview of SCORM.....</b>	<b>15</b>
<b>Chapter 3. Structuring Instruction.....</b>	<b>19</b>
<b>Chapter 4. Designing Content for SCORM .....</b>	<b>27</b>
<b>Chapter 5. Assessments and SCORM.....</b>	<b>31</b>
<b>Chapter 6. Reusable Content .....</b>	<b>37</b>
<b>Chapter 7. Tracking Learner Data.....</b>	<b>45</b>
<b>Appendix: Sources of Reusable Content .....</b>	<b>50</b>
<b>Glossary of SCORM Terms .....</b>	<b>51</b>
<b>Index.....</b>	<b>54</b>

# Detailed Table of Contents

---

<b>Chapter 1. SCORM in a Nutshell .....</b>	<b>6</b>
1.1 Introduction.....	6
1.2 What SCORM Does and Does Not Do.....	6
1.3 Highlights of SCORM 2004.....	7
1.4 SCORM Terms.....	8
1.5 Assets, SCOs, & Aggregations.....	9
1.6 Structuring Instruction .....	10
1.7 Creating Adaptive Content.....	12
1.8 Personalization.....	12
1.9 Reusable Content .....	13
1.10 The Rest of this Guide .....	14
<b>Chapter 2. Overview of SCORM .....</b>	<b>15</b>
2.1 Introduction.....	15
2.2 Example .....	15
2.3 When to apply it.....	16
2.4 What SCORM does not do .....	17
2.5 How to apply it.....	17
2.6 What's New in SCORM 2004.....	17
2.7 SCORM Programming Terminology.....	18
<b>Chapter 3. Structuring Instruction .....</b>	<b>19</b>
3.1 Introduction.....	19
3.2 Example .....	19
3.3 When to apply it.....	20
3.4 What SCORM does not do .....	20
3.5 How to apply it.....	20
3.5.1 Working with a SCORM LMS.....	21
3.5.2 Incorporating sequencing into your design .....	21
3.5.3 Flowcharting your sequencing plan .....	21
3.5.4 Creating adaptive content using sequencing rules .....	25
3.6 What's New in SCORM 2004.....	26
<b>Chapter 4. Designing Content for SCORM.....</b>	<b>27</b>
4.1 Introduction.....	27
4.2 Example .....	27
4.3 When to apply it.....	27
4.4 What SCORM does not do .....	28
4.5 How to apply it.....	28
4.5.1 Sharable content objects (SCOs) .....	28
4.5.2 What size should a SCO be? .....	29
4.5.3 Storyboards, outlining, and content design.....	29
4.5.4 Personalization (using data stored in an LMS).....	30
4.6 What's New in SCORM 2004.....	30

<b>Chapter 5. Assessments and SCORM .....</b>	<b>31</b>
5.1 Introduction.....	31
5.2 Example.....	31
5.3 When to apply it.....	32
5.4 What SCORM does not do .....	32
5.5 How to apply it.....	33
5.5.1 Assessment and SCOs.....	33
5.5.2 Single SCO .....	33
5.5.3 Multiple SCOs.....	33
5.5.4 Scoring .....	34
5.5.5 What LMS data can be used?.....	34
5.5.6 Linking Assessments to Learning Objectives .....	35
5.5.7 What's New in SCORM 2004 .....	36
<b>Chapter 6. Reusable Content.....</b>	<b>37</b>
6.1 Introduction.....	37
6.2 Example .....	37
6.3 When to apply it.....	38
6.4 What SCORM does not do .....	38
6.5 How to apply it.....	39
6.5.1 Defining reusable content .....	39
6.5.2 Creating reusable content .....	41
6.5.3 Finding reusable content.....	43
6.5.4 Metadata .....	43
6.6 What's New in SCORM 2004.....	44
<b>Chapter 7. Tracking Learner Data .....</b>	<b>45</b>
7.1 Introduction.....	45
7.2 Example.....	45
7.3 When to apply it.....	45
7.4 What SCORM does not do .....	46
7.5 How to apply it.....	46
7.6 What's New in SCORM 2004.....	49
<b>Appendix: Sources of Reusable Content .....</b>	<b>50</b>
<b>Glossary of SCORM Terms .....</b>	<b>51</b>
<b>Index.....</b>	<b>54</b>

# Chapter 1. SCORM in a Nutshell

## 1.1 Introduction

The Sharable Content Object Reference Model (SCORM) integrates a set of related technical standards, specifications, and guidelines designed to meet the high-level requirements of accessibility, reusability, interoperability, and durability of content and systems. The details of SCORM are intended for technical people like programmers. SCORM content can be delivered to learners via any SCORM-conformant Learning Management System (LMS) that uses the same version of SCORM.

Instructional designers should be familiar with the types of features SCORM supports in order to facilitate design and communicate with programmers, both of which are the purpose of this document. This document outlines SCORM 2004 version 4. This chapter presents the highlights of how to consider SCORM in instructional design.

## 1.2 What SCORM Does and Does Not Do

Table 1-1 provides a quick list of topics that SCORM informs, as well as what it doesn't inform. Some SCORM terms are included, which you'll learn more about as you read through this guide.

Table 1-1 What SCORM does and does not inform

Topic	SCORM Does Not	SCORM Does
<b>E-learning design</b>	<p>SCORM does not dictate any particular instructional design methodology, pedagogy, design pattern, or any particular organization of content.</p> <p>SCORM does not enable you, as a designer, to create e-learning that you could not create using some other method.</p> <p>SCORM is not a file format, authoring tool, or programming language.</p> <p>SCORM does not provide any guidance on any functional or instructional elements that make up and define the content, like learning objectives, assessments, or knowledge checks.</p> <p>SCORM does not dictate the formatting or look and feel of screens and the elements on them.</p>	<p>SCORM is a standard that allows you to implement your instructional design in an interoperable way, including traditional teach and test strategies as well as complex simulations.</p> <p>SCORM provides some affordances and constraints for design, but doesn't change the design process. Instructional systems designers (ISDs) will still create learning objectives that drive the design of the course. They will design content around those objectives, and then create assessments that test whether the learning objectives have been met.</p>

Topic	SCORM Does Not	SCORM Does
<b>Assessments</b>	SCORM does not take a stand on assessment content. It also does not directly address how to design and develop assessments nor when and how a SCO should be considered an assessment.	It's up to the ISD to define measurable learning objectives and to create assessments that measure those objectives whether during a test or in a simulation.
<b>Content flow</b>	SCORM does not change your ability to branch or sequence content.	SCORM allows the sequencing of SCOs (see <a href="#">Structuring Instruction</a> ) to occur through the LMS, instead of branching being hard-programmed into the content itself.
<b>Content reuse</b>	SCORM does not dictate the ways in which content should be reused. SCORM also does not require that all content be reused.	SCORM enables the reuse of content by defining standards for how content is structured (see <a href="#">Reusable Content</a> ) and packaged.
<b>Performance data</b>	SCORM does not dictate what kind of information you must collect, or how it must be used.	SCORM gives you the ability to track and access certain kinds of information (see <a href="#">Tracking Learner Data</a> ) about your SCO.
<b>Development tools</b>	SCORM does not recommend any specific tool, programming language, system, or learning technology architecture. SCORM also does not define the additional features an LMS may have, such as authoring, classroom management, competency management, knowledge management, certification or compliance training, testing, personalization, mentoring, video conferencing, chat, and discussion boards.	SCORM simply defines the structure that the content you are placing inside an LMS should have, and how the LMS delivers that content. SCORM defines a technical organization, not an instructional organization. This technical organization is typically taken care of by the programmer or authoring system.

## 1.3 Highlights of SCORM 2004

Table 1-2 highlights the important features of SCORM 2004 as they've changed from SCORM 1.2:

Table 1-2 SCORM 2004 features

Topic	Description
<b>Sequencing</b>	SCORM 2004's sequencing rules allow instructional designers to specify the order in which SCOs are delivered to learners in a SCORM 2004-conformant LMS. Prerequisites to SCOs can also be specified to ensure that learners know certain information or can complete a particular SCO before moving to another subject or SCO. Remediation can be specified based on learners' performance within a particular SCO. Sequencing is addressed in more detail in <a href="#">Structuring Instruction</a> .
<b>Data Model Elements</b>	SCORM data model elements define the types of information that can be stored in an LMS. With SCORM 1.2, LMS programmers could choose to implement only a subset of the SCORM data model elements, which contributed to interoperability problems. SCORM 2004 standardized the data model elements and now requires all LMSs to implement them to achieve SCORM 2004 conformance. There are now also additional data model elements, and the number of characters of data that can be saved as increased. This enhances the amount of customization and personalization that you are able to incorporate into your design. SCORM data model elements are addressed in more detail in <a href="#">Tracking Learner Data</a> .

Topic	Description
<b>Metadata tags</b>	Metadata tags changed somewhat from SCORM 1.2 to SCORM 2004. These changes were made primarily to incorporate industry standards into the SCORM model. Your programmer is likely aware of these changes.

## 1.4 SCORM Terms

Table 1-3 lists SCORM terms as they relate to common ISD terms. These are useful to know in order to communicate well with programmers who will be implementing your courses. Some commonly used SCORM or ISD terms have official terms used in the SCORM Books (technical guides for programmers) that you should also be aware of.

Table 1-3 ISD and related SCORM terms

COMMON ISD TERM	COMMON SCORM TERM	OFFICIAL SCORM TERM
Video, text file, image, or other media; also called asset	Asset	Asset; Also, a Resource is a group of assets
Learning object (LO)	Sharable Content Object (SCO)	SCO
(None)	Activity: a SCO or logical grouping or aggregation of SCOs, with associated sequencing and rollup rules. Can represent specific instructional strategies. Activities can be embedded within activities.	Activity
Organizational Structure	An Activity Tree represents the data structure that an LMS implements to reflect the hierarchical, internal representation of the defined learning activities.	Activity Tree
Course content (all assets, LOs, branching, structure for a course)	Content Package (a zip file that contains the course content)	Content Package
(None)	Content Aggregation represents the collection of content and its structure represented within a Content Package	Content Aggregation is a nested structure. A content package might have 1 or 100 aggregations.
Branching	Sequencing (not internal branching within a SCO)	Sequencing
Scoring	Objective (NOTE: this is not related to the ISD term Learning Objective)	Objective
Branching instructions to the programmer, flowcharts	Rollup and Sequencing rules (not internal branching within a SCO)	Rollup and Sequencing rules
Learner interactivity data (any interactions with content, usually used for assessment data)	Interactions	Interaction Data/ cmi.interactions

## 1.5 Assets, SCOs, & Aggregations

The SCORM organizational structure is composed of several components: Assets, SCOs, Aggregations, and Organizations. We start with some definitions.

An *asset* refers to a video, text file, image, sound file, HTML page, assessment object, or other pieces of content. Assets may end up being your most reusable items that can be redeployed, rearranged, repurposed, or reused in many different contexts and applications. Assets are the pieces of data that a SCO is built around.

*Sharable Content Objects*, or *SCOs*, are units of information you can deliver to your learners via an LMS, and are the smallest unit of information that can communicate with the LMS to track student performance or interactions. They are often roughly equivalent to lessons, learning objects such as simulations or other complex activities, instructional methods that stand as a unit, and modules, although they can be, and often are, smaller “chunks” of instruction.

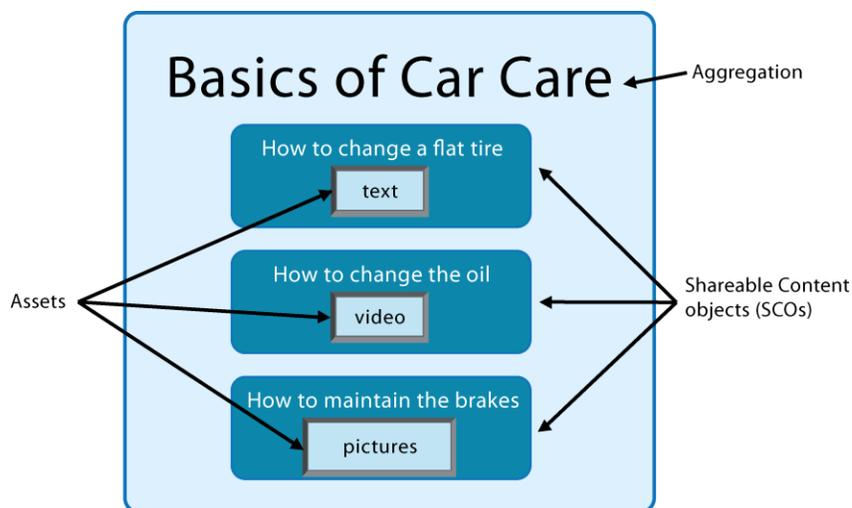


Figure 1-1 Basic SCORM Aggregation with SCOs and assets

An *aggregation* can be defined as a collection of related SCOs. You could consider an aggregation roughly equivalent to a course or section of a course. The top-level aggregation of the content structure you have designed is sometimes called the *organization*. Programmers also refer to it as the root aggregation.

Now for an example: An ISD typically designs a course based on multiple learning objectives. For example, a course on basic car care may have modules pertaining to three learning objectives: how to repair a flat tire, how to change the oil, and how to maintain the brakes. In SCORM, this course (as shown in Figure 1-1) might be structured as an aggregation containing three SCOs (discussed in more detail in [Sharable content objects \(SCOs\)](#)), one for each of the learning objectives, and each SCO containing one or more content assets.

Aggregations can be fairly simple, like the one shown in Figure 1-1, but they can also be quite complex, depending on the needs of your design. You can have aggregations inside other aggregations, if that's the design you require. Figure 1-2 shows a more complex aggregation structure.

In this example, Basics of Car Care is a smaller aggregation that has been combined with other SCOs inside another aggregation called Maintaining your Car. Remember that each SCO can be any type of learning object that you want, from simple text and pictures, to animations or complex simulations.

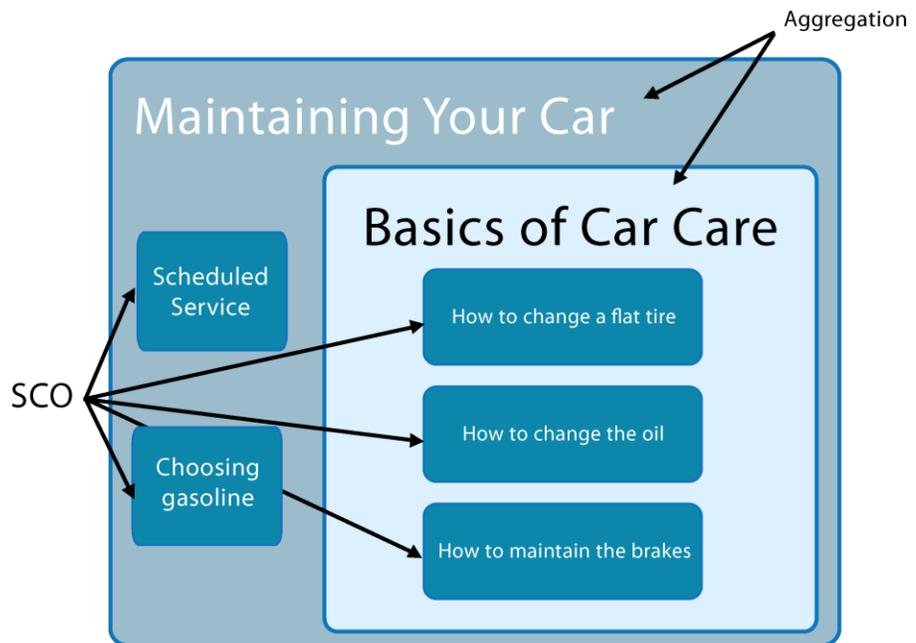


Figure 1-2 Aggregation Structure

This type of design keeps things modular and therefore reusable. And it also makes it easy to do SCORM-conformant sequencing.

## 1.6 Structuring Instruction

You may currently use branching in your e-learning courses. Typically, branching is hard-coded inside the content, or inside the LMS. This limits interoperability across LMSs. With SCORM 2004's sequencing rules, you can dictate branching (what SCORM calls "sequencing") between SCOs. When you do that, it's easy for the same course to be used in any SCORM 2004-conformant LMS.

To begin establishing the sequencing rules, you will need to make some design decisions. Going back to our car example, we'll start with deciding how learners will move from one SCO to the next. Can learners take them in any order? Do they have to pass one section before taking another? What about after they have finished each SCO – can they return to a SCO in order to review it?

SCORM in a Nutshell

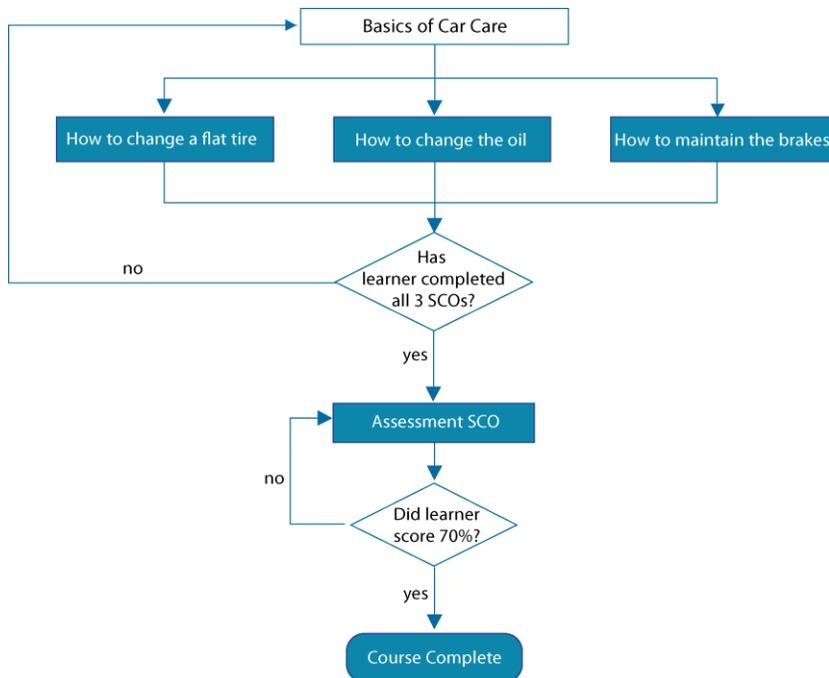


Figure 1-3 Sequencing a course with an assessment

Figure 1-3 shows a flowchart depicting one way to sequence the Basics of Car Care course (as originally shown in Figure 1-1), including an assessment. In this course design, the learner can choose the SCOs in any order, but must review all of them before moving to the assessment. Also, the learner can take the assessment until at least 70% is achieved before the course is considered completed.

In another course design option, shown in Figure 1-4, the three SCOs are designed to be taken in a particular order, so we don't have to check to see if they were all completed.

Flowcharts like these should be used to share the flow of your course design with your programmers so they can write the sequencing rules to implement your design.

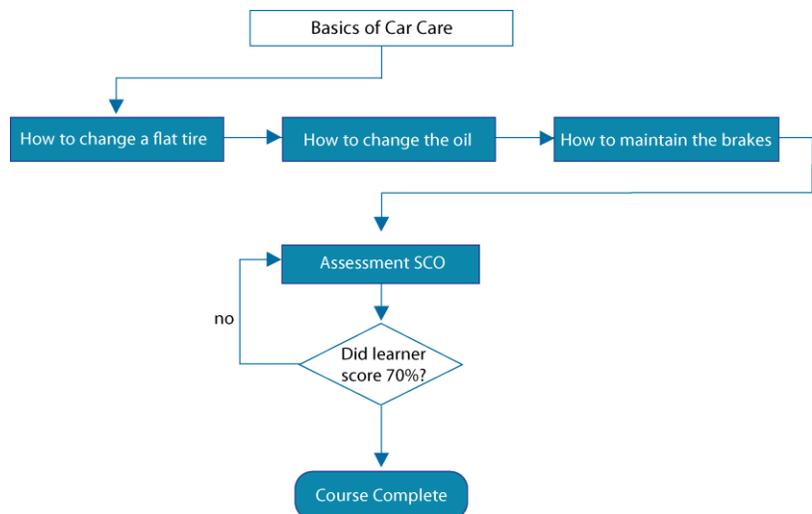


Figure 1-4 Sequencing SCOs in particular order

## 1.7 Creating Adaptive Content

You can use sequencing to create adaptive content. For example, you can offer a pretest and base the sequencing on the results of that pretest, sending learners only to the topics where they need more work. Similarly, you can offer remediation after a post-test by returning users to SCOs, or modules, where they need more work. See [Assessments and SCORM](#) of this guide for more details on this.

Figure 1-5 shows remediation after a post-test. Of course, as the instructional designer you would determine what constitutes a passing score for each learning objective in the assessment.

SCORM will not tell you how to storyboard, outline, or design your content, although you *do* need to make sure your storyboards, outlines, and flowcharts fully explain how you want your content to be structured.

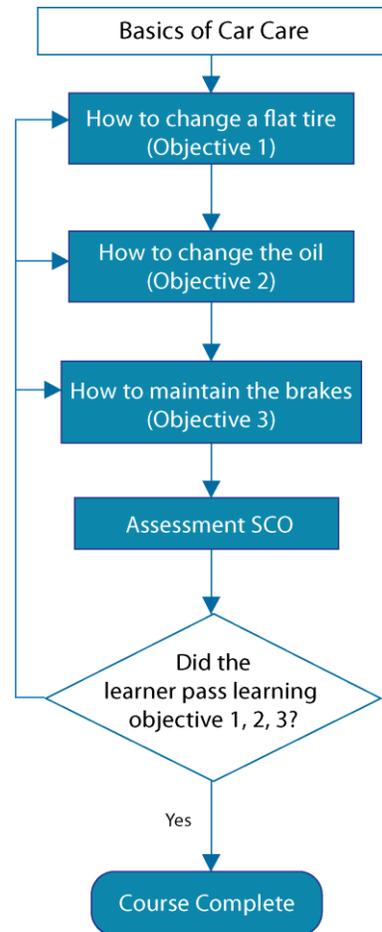


Figure 1-5 Remediation

## 1.8 Personalization

One thing that SCORM data can do to enhance your content design is to allow for personalization of the content in your SCOs. Certain data can be sent to the LMS from the SCO, and likewise, certain data can be sent to the SCO from the LMS.

For example, you might want to retrieve information from the LMS to customize your learners' experience such as:

- The learner's name for use inside the content
- The last location in the content the learner viewed, or bookmarking
- The learner's language or presentation options (e.g., transcript turned on, audio off)

Other data available to you include:

- Score
- Total time spent in a SCO
- Time spent in a single session of a SCO
- Completion status
- Responses to assessment items
- Interactions within a SCO
- Pass/fail status

This personalization occurs through what SCORM calls Data Model Elements, which allow the LMS to collect data about learner progress in the SCO (see [Tracking Learner Data](#) for more details).

## 1.9 Reusable Content

Any time you are creating content, you should take reuse into consideration. First, see if there is existing content that can be reused. Also consider the reusability of your own content as you are designing and implementing it.

The first step in creating reusable content is making it granular. This means that SCOs should be the smallest size possible for your purposes. To determine how best to divide or chunk your content, look at the potential audiences you have identified and ask yourself the following questions:

- What portions of this content apply only to my target audience?
- Can the content be divided before and after the portion that applies only to my target audience?

Another key to keeping content reusable is making it context-neutral. This means that a particular SCO can be separated from its package and still be considered complete—that is, understandable to the learner without the need to refer to critical information that is not in the SCO. Where context-specific instruction is required, you can create context-specific SCOs to address those objectives.

To find reusable content, there are many avenues you can take—you can look in your own library, search Google, ask friends in Facebook, LinkedIn, or other social networks, and search organizational repositories. A detailed list of sources of reusable content may be found in an appendix.

Finally, to make your content reusable, you should tag it with appropriate metadata. Metadata is the information that describes your content, both the individual components (the assets and SCOs) as well as the entire course (which SCORM calls content packages).

## 1.10 The Rest of this Guide

The remaining chapters give more details about each of the topics introduced above. Happy reading!

### **Best Practices:**

When designing content to be SCORM-conformant, ask yourself these questions:

- 🔔 How do you want your content to be structured and presented to your learners?
- 🔔 Will SCOs cover a single learning objective or multiple learning objectives or will it vary as needed for the project?
- 🔔 How will SCOs be divided, structured, and sequenced?
- 🔔 Will SCOs include an embedded assessment, or will the assessment be a separate SCO?
- 🔔 What kind of data do you need to track in the LMS?
- 🔔 How will you maximize the potential for content to be redeployed, rearranged, repurposed, and rewritten? For example, can you use templates or cascading style sheets to address reusability?

## Chapter 2. Overview of SCORM

### 2.1 Introduction

The ADL Initiative was established in 1997 to standardize and modernize training and education management and delivery. The vision of the ADL Initiative was to provide standards and technology that would enable the Department of Defense (DoD) to have access to the highest-quality learning and performance support that could also be tailored to individual needs and delivered cost-effectively, at the right time and in the right place.

To meet its mission, the ADL Initiative developed the Sharable Content Object Reference Model (SCORM). SCORM integrates a set of related technical standards, specifications, and guidelines designed to meet SCORM's high-level requirements of accessibility, reusability, interoperability, and durability of content and systems. SCORM content can be

"ISDs with no e-learning experience might have a hard time understanding SCORM. However, SCORM does not really change things for the ISD with experience in e-learning."

— SCORM 2004 Expert

delivered to learners via any SCORM-conformant Learning Management System (LMS) that uses the same version of SCORM. You may be familiar with, or may hear references to SCORM 1.2, which is an earlier version of SCORM. The current version, SCORM 2004, introduced a new set of capabilities that were not available with SCORM 1.2. SCORM 2004 includes editions that have improved the 2004 specification slightly over the years (without changing the basic framework and capabilities of 2004). At the time of this writing, SCORM 2004 4<sup>th</sup> Edition is the current version of SCORM. This guide is an introduction to SCORM 2004 (all editions) for the instructional systems designer with e-learning experience, and will also help ISDs with SCORM 1.2 experience move to SCORM 2004.

### 2.2 Example

Table 2-1 provides a brief explanation and example for each of the four architectural functionality requirements addressed by SCORM. These are sometimes referred to as the ADL "ilities".

Table 2-1 SCORM High-Level Requirements

Requirement	Explanation	Example
<b>Accessibility</b>	Content can be located and accessed from multiple locations and delivered to other locations.	A content author can search an organizational repository, and identify relevant content that has already been developed, possibly even by another organization, and deploy that content on an LMS to learners anywhere in the world.
<b>Interoperability</b>	Content operates across a wide variety of hardware, software, operating systems, and web browsers regardless of the tools used to create it and the platform on which it was initially delivered.	Content packaged for delivery in one SCORM-conformant LMS can be loaded into another SCORM-conformant LMS for delivery to learners.
<b>Durability</b>	Content does not require modification to operate as versions of software systems and platforms are changed or upgraded.	Upgrades to local computers, such as changing to a new computer operating system, or server-side changes, such as upgrading to new versions of the LMS should have no impact on the delivery of content to learners.
<b>Reusability</b>	Content is independent of learning context and is able to stand alone. It can be used in numerous training situations or for many different learners.	E-learning content designed for one organization can be easily redeployed, rearranged, repurposed, or rewritten for or by other organizations that have similar learning needs.

## 2.3 When to apply it

SCORM only applies to web-based e-learning delivered from an LMS or using an LMS service. It does not apply to CD-ROM or file server-based e-learning, classroom learning materials, and most immersive learning environments.

You, your client, or your project manager need to determine whether (a) you need to create web-based e-learning, (b) it is a client requirement that the content you create be SCORM-conformant, or whether you want the e-learning you create to be interoperable via SCORM, and (c) whether the LMS you will be deploying it on (or LMS service you are using) is SCORM conformant (ideally, SCORM certified). If any of these conditions are not met, you do not need to apply SCORM to your course.

In regards to (b), it should be noted that if you are designing e-learning content for the U.S. DoD, it is required that your content be conformant to the current version of SCORM. This is in accordance with [DoD Instruction 1322.26](#) (this Instruction is currently under revision—however, this particular requirement is not expected to change). This instruction also states that any LMS deployed in DoD must be conformant to the current version of SCORM.

It would be beneficial to first understand what SCORM offers before beginning the design of your course. This will allow you to incorporate any applicable SCORM features in your course design.

## 2.4 What SCORM does not do

SCORM does not provide any guidance on any functional or instructional elements that make up and define the content, like learning objectives, assessments, or knowledge checks, and it does not dictate the formatting or look and feel of screens and the elements on them. Also, it does not dictate any particular organization of content information. All of these design decisions are still completely up to the instructional designer.

SCORM does not dictate a particular instructional design approach, but it does provide some affordances and constraints for design. Above all, it is important to understand that SCORM does not enable you, as a designer, to create e-learning that you could not create using some other method. It is not a file format, authoring tool, or programming language. It is a collection of standards (or, technically speaking, a reference model) that allows you to implement your instructional design in an interoperable way.

## 2.5 How to apply it

SCORM provides technical standards for the structure of e-learning courses. The purpose of these standards is to facilitate interoperable reusability and individualization that SCORM affords. ISDs need to pay attention to how to structure SCORM learning modules (aka Sharable Content Object or *SCO*; described in [Assets, SCOs, & Aggregations](#)). This will ensure that programmers have the information they need to code the e-learning design correctly. Further details are provided in the respective sections about what an instructional designer needs to consider to make e-learning SCORM-conformant.

## 2.6 What's New in SCORM 2004

The current version, SCORM 2004, introduces a new set of capabilities that were not available in the previous version, SCORM 1.2, and these changes are presented in Table 2-2. Note that not all organizations have upgraded to SCORM 2004. Therefore, before beginning any SCORM content development project, you must ensure that the target LMS is SCORM 2004-certified in order to apply the concepts addressed in this document. Furthermore, you need to determine to which edition of SCORM 2004 the LMS is certified; backward compatibility (for example, a 4<sup>th</sup> Edition-certified LMS running 3<sup>rd</sup> Edition content) is not generally a problem, but forward compatibility usually is.

Table 2-2 Changes from SCORM 1.2 to SCORM 2004

Topic	Description
<b>Sequencing</b>	SCORM 2004's sequencing rules allow instructional designers to specify the order in which SCOs are delivered to learners in a SCORM 2004-conformant LMS. Prerequisites to SCOs can also be specified to ensure that learners know certain information or can complete a particular SCO before moving to another subject or SCO. Remediation can be specified based on learners' performance within a particular SCO. Sequencing is addressed in more detail in <a href="#">Structuring Instruction</a> .
<b>Data Model Elements</b>	SCORM data model elements define the types of information that can be stored in an LMS. With SCORM 1.2, LMS programmers could choose to implement only a subset of the SCORM data model elements, which contributed to interoperability problems. SCORM 2004 standardized the data model elements and now requires all LMSs to implement them to achieve SCORM 2004 conformance. There are also additional data model elements, and the number of characters of data that can be saved has increased. This enhances the amount of customization and personalization that you are able to incorporate into your design. SCORM data model elements are addressed in more detail in <a href="#">Tracking Learner Data</a> .
<b>Metadata tags</b>	Metadata tags changed somewhat from SCORM 1.2 to SCORM 2004. These changes were made primarily to incorporate industry standards into the SCORM model. Your programmer is likely aware of these changes.

## 2.7 SCORM Programming Terminology

Throughout this guide, you will be introduced to terms your programmer may use when discussing how to implement the SCORM guidelines in your course. These terms are explained in the [Glossary](#) of SCORM Terms.

## Chapter 3. Structuring Instruction

### 3.1 Introduction

When you introduce SCORM, you do not change the process of designing instruction. However, some aspects of your design process may have particular requirements in response to SCORM standards. The content you create can be as simple as basic linear movement between SCOs or as complex as elaborate branching. You are probably familiar with, and have designed branching content. The content may remediate learners back to sections where they need more work, based on their scores. Another course may present different material to different learners based on certain variables, such as their performance on a pre-test. SCORM 2004 defines an interoperable way to implement these kinds of structures; however, it may require you to think about them in a different way. This section covers the flexibility SCORM provides for sequencing your instruction to fit your needs.

### 3.2 Example

Let's return to the Car Care course we discussed in Section 1, shown again in Figure 3-1. The basic course has 3 SCOs. But now decisions must be made about how learners will move from one SCO to the next. Can learners go through them in any order? Do they have to pass one section before taking another? What about after they have finished each SCO – can they return to a SCO in order to review it?

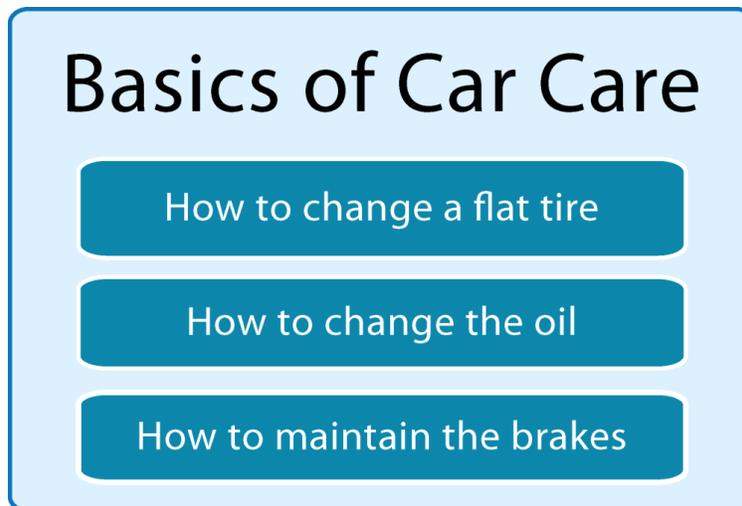


Figure 3-1 Basics of Car Care SCOs

**Best Practice:** A major benefit of a SCORM course is that it is easily portable to a different SCORM-conformant LMS.

### 3.3 When to apply it

All content design requires you to think about the path learners will take through the content. SCORM sequencing rules simply allow you to set that path in an interoperable way to order the delivery of content. Because the sequencing is not dependent on the LMS, it also makes it easy for you to reuse or repurpose content.

### 3.4 What SCORM does not do

SCORM does not change your ability to sequence content. What SCORM does is change the way such content is programmed. Instead of the branching being hard-programmed into the content itself, SCORM allows the sequencing of SCOs to occur based on rules that you define, as discussed in this section.

### 3.5 How to apply it

Sequencing allows you to control what content is presented to learners and when it is presented. Sequencing rules reside in the manifest file (see [Glossary of SCORM Terms](#) for definition), which is a part of the SCORM content package (also found in the Glossary), so

content can be sequenced without relying on hard-coded data inside a SCO or a particular LMS.

### 3.5.1 Working with a SCORM LMS

Sequencing rules are based on standardized behaviors (defined for programmers in the SCORM Sequencing and Navigation book) that all SCORM 2004-conformant LMSs must support. Essentially, the LMS reads the sequencing rules from the manifest file, locates the appropriate SCO to deliver to the learner, launches the SCO in the learner's browser, collects data about learners' performance and status, and then processes the next set of sequencing rules from the manifest. These are all things that your programmer helps facilitate and implements to your specifications. So, while you don't have to understand all the details, it's useful to understand the general approach to help you to communicate well with your programmer.

### 3.5.2 Incorporating sequencing into your design

Structuring your content for sequencing does not change your design process. The decisions you make about how the content is delivered to students form the basis for the sequencing rules your programmer will create. A good time to start considering these questions is when you are outlining your content:

- What are your assessment and remediation requirements?
- Will learners be able to choose the content they see, or will you prescribe the order?
- Do you want to adapt the learner's experience based on their choices or decisions?
- How will you determine when or whether the learner has completed or passed the content?

### 3.5.3 Flowcharting your sequencing plan

Since your programmer will be the one actually incorporating the sequencing rules into the SCORM content package, you will need to provide a detailed description of the navigation and/or sequencing paths you desire your content to have. This may require an extra step in your content creation process: a flowchart or flow diagram which provides this detail. The flowchart is separate from your content outline, which specifies the individual pieces of content you plan to include, or your storyboards, which detail the screen-by-screen content to be contained inside your SCOs (see [Assets, SCOs, & Aggregations](#)).

Let's return to our example, the Basics of Car Care course. The sequencing decisions we made include the following:

## Structuring Instruction

- Learners may take the three SCOs, or sections, in any order they wish.
- When learners have finished all three SCOs, they will take the assessment.
- Learners must score 70% or above to pass the assessment and complete the course.
- If learners score less than 70%, they will retake the assessment. There is no limit on the number of times they may retake the assessment.

Figure 3-2 shows what the flowchart of these rules might look like. Notice that the flowchart uses arrows to show exactly the order in which things should happen. There are decision points (shown in the diamonds) to express the conditions that have to be met for the sequence to progress. These decision points are what your programmer will use to create the sequencing rules.

Structuring Instruction

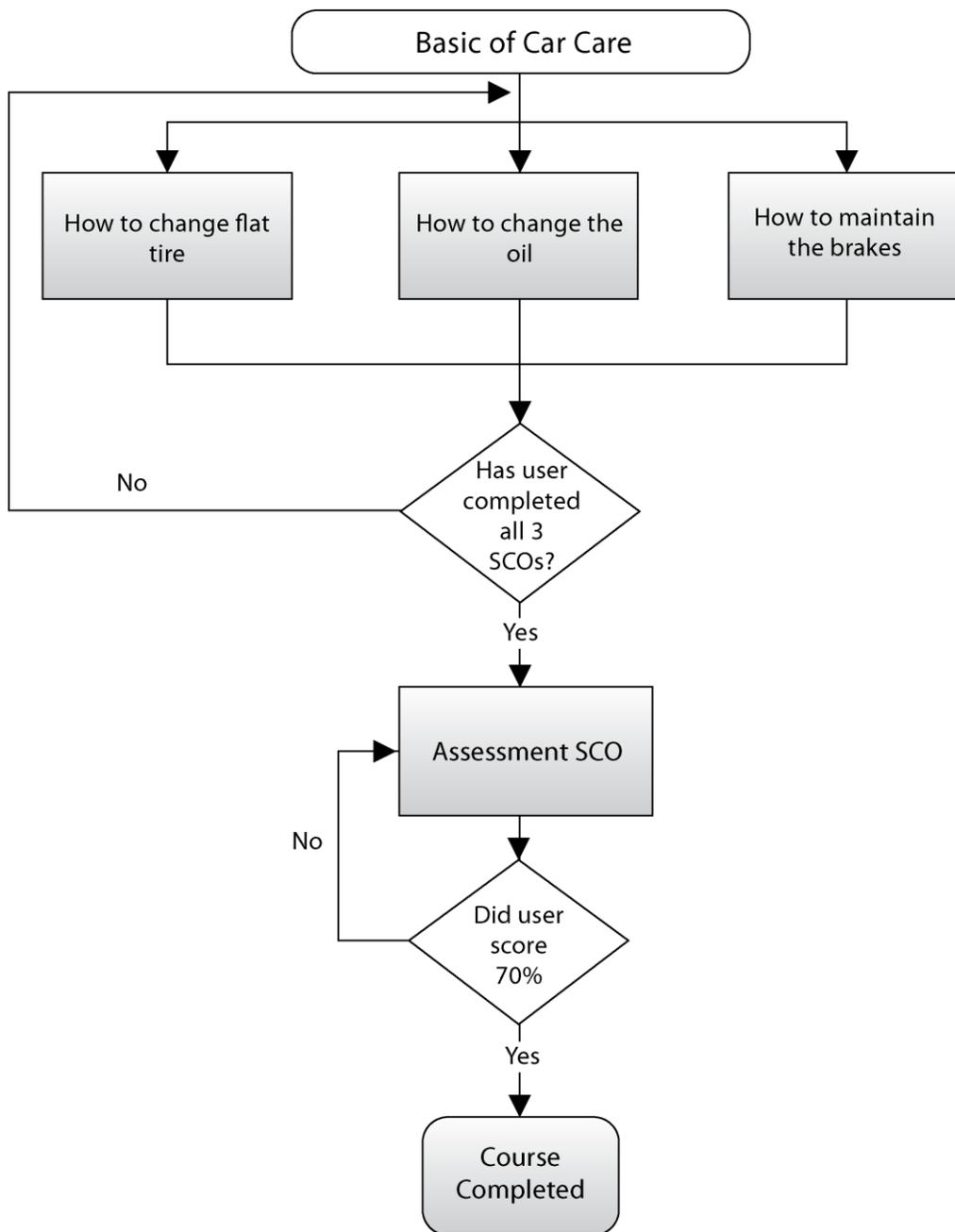


Figure 3-2 Basics of Car Care Sequencing Flowchart

You can also include sequencing rules between aggregations (see [Assets, SCOs, & Aggregations](#)), which adds to both the adaptability of your content, and to the complexity of your design. Let's look at our Car Ownership aggregation (Figure 3-3).

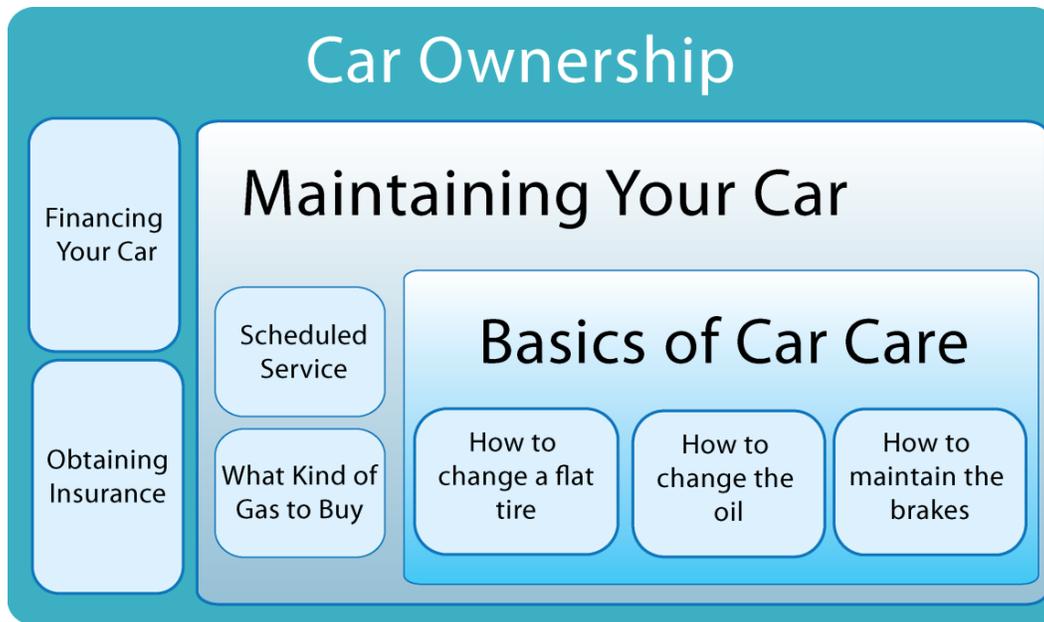


Figure 3-3 Car Ownership Aggregation

In your flowchart, you would want to specify how the learner should proceed, not only from SCO to SCO, but also between and within the aggregations. Figure 3-4 shows what a flowchart for this aggregation might look like.

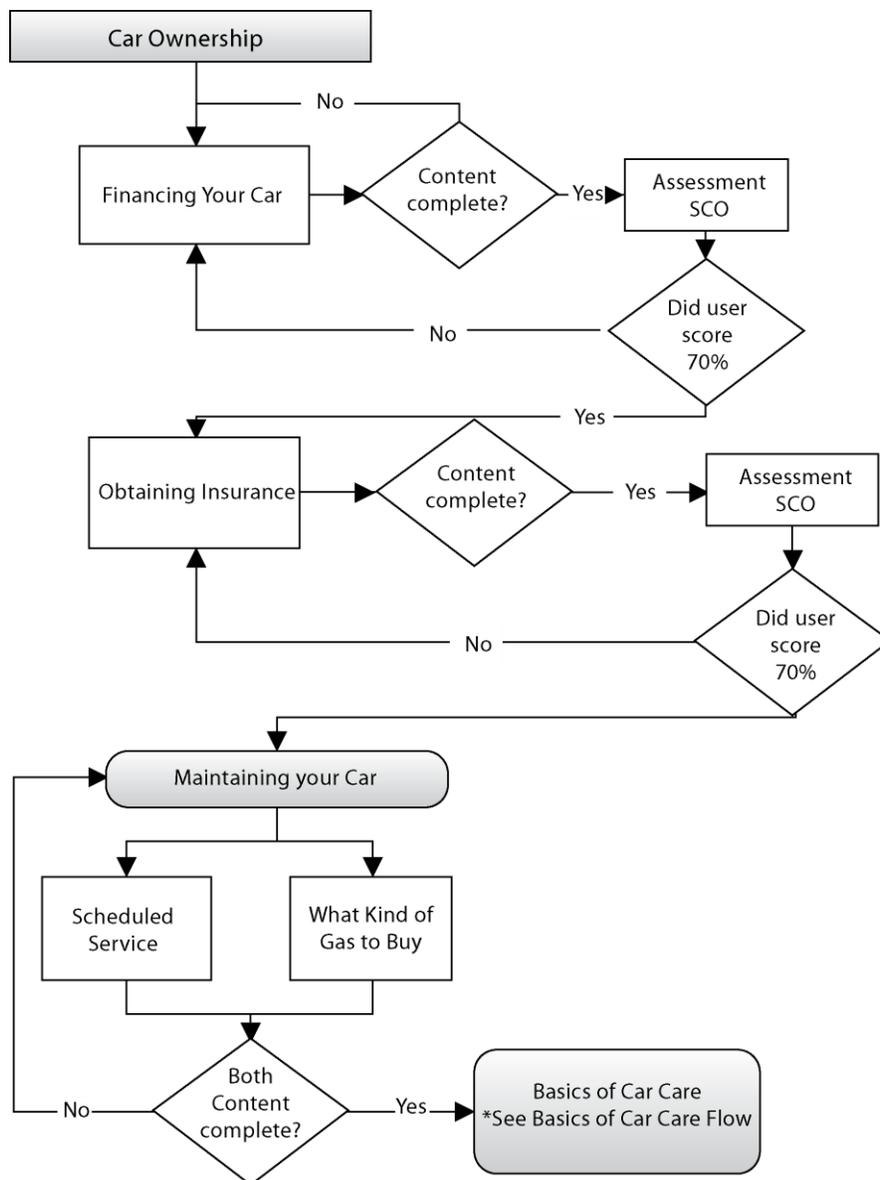


Figure 3-4 Car Ownership Aggregation Sequencing Flowchart

### 3.5.4 Creating adaptive content using sequencing rules

The sequencing rules you specify can also be used to create adaptive content. You can offer a pretest, and base the sequencing on the results of that pretest, sending learners only to the topics where they need more work. In the same way, you can offer remediation after a post-test by returning users to SCOs, or modules, where they need more work. See [Assessments and SCORM](#) of this guide for more details on this.

Many other options for the design of your adaptive content are available by using combinations of data model elements, sequencing rules, and global objectives (the latter is

a technical term—do not confuse it with learning objectives), all the way up to and including highly dynamic simulations with interaction nodes that determine the learner's experience based on performance and/or choices taken.

## 3.6 What's New in SCORM 2004

In SCORM 1.2, creating content that either contained a specified navigation path or branching required the use of large SCOs containing the required programming within the SCO. This makes the content less reusable. The sequencing rules in SCORM 2004 allow ISDs to prescribe the manner in which learners receive individual pieces of content from the LMS, regardless of which LMS is used. As a result, you can design more granular content and allow the LMS to control the movement of learners from SCO to SCO in accordance with the behaviors you specify; this increases the possibility that your SCOs will be reusable because there are no hard-coded rules in your SCOs to modify.

# Chapter 4. Designing Content for SCORM

## 4.1 Introduction

The primary responsibility of an instructional designer is designing effective instructional materials. SCORM does not change that role. Instructional designers and content authors do not need to master all of the technical nuances of a SCORM implementation. SCORM simply introduces technical guidelines that provide a framework in which you can work to create content that will work within any SCORM-conformant LMS as well as be available for reuse. This section will talk about how to accommodate these guidelines in your content design process. In creating SCORM-conformant content, you will need to take into account and plan for certain SCORM elements with your design, but these are meant to enhance your content, not limit it.

## 4.2 Example

Consider a course that provides learners with broad instruction about handling hazardous materials for compliance issues. The only requirement is to show that learners have completed the content. In this case, completing the content usually means learners have viewed it, so the content for this course may be a single SCO, which then communicates the completion to the LMS where the course resides.

In a second scenario, the requirement may be that the learners have completed several specific elements of the hazardous materials course, such as “how to properly load and store radioactive materials during transit” and “how to properly maintain records of hazardous materials being transported.” In this case, the course may be composed of several SCOs, each of which contains content for the specific elements. Each of these SCOs would then communicate to the LMS, so that the completion of the individual elements can be tracked.

## 4.3 When to apply it

SCORM considerations should be taken into account throughout your design process. When analyzing content, you should consider reusing existing materials which you may be able to find in content repositories. In design and/or development phases, you need to be aware of what SCORM can do to enhance your content delivery, such as allowing for remediation following assessments (see [Assessments and SCORM](#) for further details), allowing

personalization of the content (see [Personalization \(using data stored in an LMS\)](#)) for further details), or accommodating the bookmarking of content as the learner goes through the course.

As design-related questions arise—such as, “Can I do this with the content and still make it SCORM conformant?”—you will rely on the programmers’ technical expertise to explain the technical constraints under which they are working. Likewise, as programmers begin the technical implementation of the instructional materials, they will need to confer with you to ensure that the materials are functioning in a SCORM-conformant LMS as you intended. It is essential that ISDs and programmers form a cohesive team that can work together from the initial planning stages of the project through delivery.

## 4.4 What SCORM does not do

SCORM does not dictate any particular design methodology or design pattern. The instructional systems designer will still create learning objectives that drive the design of the course. They will design content around those objectives, and then create assessments that test whether the learning objectives have been met.

## 4.5 How to apply it

### 4.5.1 Sharable content objects (SCOs)

SCOs are the smallest logical unit of information you can deliver to your learners via an LMS. The term SCO has different implications for instructional designers and programmers. Instructional Systems Designers (ISDs) and content authors view a SCO as content; they focus on the actual instructional material in the SCO. Programmers may view a SCO as a location in a Table of Contents for which to apply programming rules, or as a collection of assets that will communicate with the LMS.

From an ISD perspective, a SCO is most commonly a defined piece of instruction comprised of one or more assets (see [Assets, SCOs, & Aggregations](#)). Since SCOs collect information on learners’ performance, structure SCOs based upon the specific performance needs of your learners. One focus of SCORM is to design courses so that SCOs are context-independent and of a reasonable grain-size so they can be reused.

Assessment SCOs should report a completion status (e.g., incomplete or completed) and some type of result (e.g., pass/fail or a number). Pass/fail can also be generated

automatically from a numeric or percentage score. You decide what constitutes completion and a result, and should inform your programmer about this decision.

## 4.5.2 What size should a SCO be?

The ISD determines what size a SCO should be, based on the design. A simple single-SCO course with multiple objectives, like the one described in the example above, could be very large. A SCO that is found inside a larger course, might be very small – a single HTML page can be designated a SCO if it needs to communicate with the LMS.

**Best Practice:** SCO size is dependent on reuse and learner performance tracking requirements. If you need to capture lots of learner information, that will influence the size of your SCOs. If reuse is important, SCOs will have to be designed according to your best estimate of the modules that can be reused.

## 4.5.3 Storyboards, outlining, and content design

SCORM will not tell you how to storyboard, outline, or design your content, although you *do* need to make sure your storyboards, outlines, and flowcharts fully explain how you want your content to be structured. As you are preparing for each phase of your design process, you should keep some SCORM elements in mind.

When designing content to be SCORM conformant, consider the following:

- How do you want your content to be structured and presented to your learners?
- Will SCOs cover a single learning objective or multiple learning objectives or will it vary as needed for the project?
- How will SCOs be divided, structured, and sequenced?
- Will SCOs include an embedded assessment, or will the assessment be a separate SCO?
- What kind of data do you need to track in the LMS?
- How will you maximize the potential for content to be redeployed, rearranged, repurposed, and rewritten? For example, can you use templates or cascading style sheets to address reusability?

## 4.5.4 Personalization (using data stored in an LMS)

One thing that SCORM data can do to enhance your content design is to allow for personalization of the content in your SCOs. Certain data can be sent to the LMS from the SCO, and likewise, certain data can be sent to the SCO from the LMS.

For example, you might want to retrieve information from the LMS to customize your learners' experience such as:

- The learner's name for use inside the content (i.e., "Well done, Jane.")
- The last location in the content the learner viewed (i.e., "Do you want to start where you left off?")
- The learner's language, presentation (e.g., transcript turned on, audio off), or other preferences

You may also want to store information in the LMS such as the learner's:

- Score
- Total time spent in a SCO
- Time spent in a single session of a SCO
- Completion status
- Responses to assessment items
- Interactions within a SCO
- Pass/fail status

This personalization occurs through Data Model Elements, which allow the LMS to collect data about learner progress in the SCO (see [Tracking Learner Data](#) for more details). The programmer controls exactly how these things are communicated back and forth from the LMS to the SCO, but as the content designer, you need to be aware of what is available, so you can take advantage of its availability.

## 4.6 What's New in SCORM 2004

SCORM 2004 adds sequencing and navigation capabilities that give you more freedom to design your content as you need it to be. In addition, SCORM 2004 adds additional data model elements and increases the number of characters of data that can be saved, which enhances the amount of customization and personalization that you are able to incorporate into your design.

# Chapter 5. Assessments and SCORM

## 5.1 Introduction

Planning for assessments is a key part of any design process. Questions about whether to provide a pretest and what will be covered in the posttest are important decisions that often get made early. This section will talk about creating SCORM-conformant assessments, whether they are pretests or posttests. It will also talk about SCORM-conformant options that are connected to taking an assessment, like scoring, feedback, and remediation.

To SCORM, an assessment is just another SCO or set of SCOs (see [Designing Content](#)). The ISD will decide whether it is a single SCO (see [Sharable content objects](#) (SCOs)) (a single test with many items) or multiple SCOs (one SCO per test item).

SCORM also facilitates the delivery of performance reports, allows the matching of performance to learning objectives, and provides for the use of remediation and feedback to further individualize learning experiences.

**Best Practice:** Most LMSs provide their own assessment authoring modules, with which you can create assessments that can be made to work with your course. However, you should avoid them if you want the assessment to be portable outside of the LMS with your content. Even if the assessment can be exported, you will still want to make sure it is SCORM-conformant.

## 5.2 Example

A multiple-choice test is assessing knowledge on three objectives (Objective 1, Objective 2, and Objective 3, shown in Figure 5-1). On question 1, which is concerning Objective 1, a learner might incorrectly select “A” to a question where the correct response is actually “C”. Selecting “A” as the answer seems to indicate that the learner has certain misconceptions about that Objective. Learning can be individualized by providing specific feedback to the learner as well as remediation by sending the learner to a specific SCO that addresses the weakness or misconception.

An assessment can be given at any time in the course, for example, a pretest, an end-of-module, or end-of-course test. A pretest can be set up so that learners can be directed to the specific lessons (SCOs) for which they are lacking knowledge. An end-of-module or end-of-course test would provide remediation by redirecting the learner back to the lessons (SCOs) indicated by the test results.

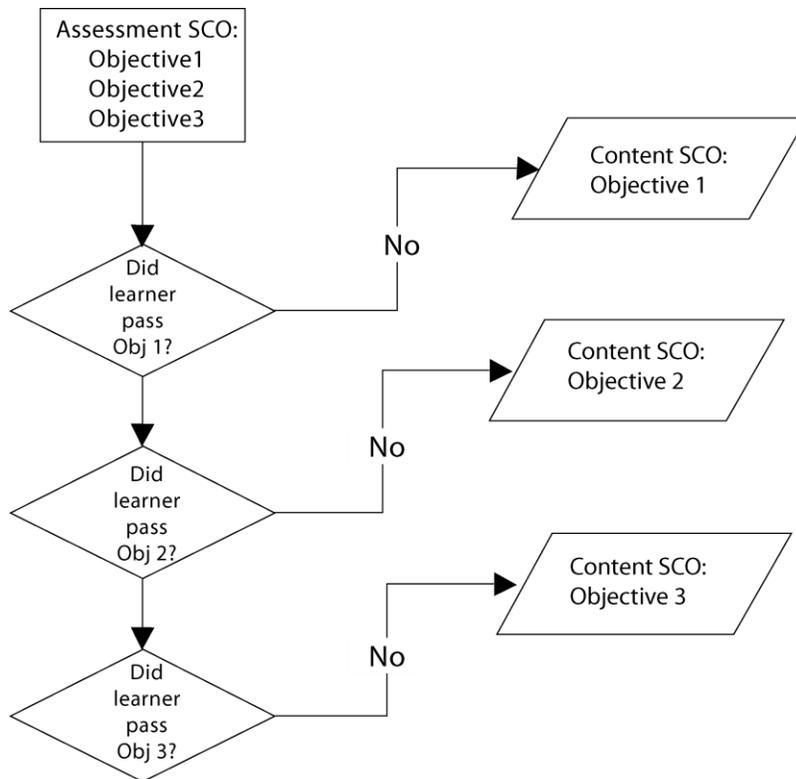


Figure 5-1 Flowchart showing Assessment with Remediation

### 5.3 When to apply it

A SCORM-conformant assessment should be used when you want to allow learners to “test out” of portions they have already mastered, save performance data, or provide remediation (see [Creating adaptive content using sequencing rules](#)).

### 5.4 What SCORM does not do

SCORM does not take a stand on assessment content. It’s up to the ISD to define measurable learning objectives and to create assessments that measure those objectives.

SCORM does not directly address how to design and develop assessments nor when and how a SCO should be considered an assessment.

**Best Practice:** Knowledge checks can be used to give formative feedback to the user with or without being scored. In that case, the knowledge check will sit inside a single SCO along with the content it’s addressing, and does not need to communicate SCORM data to the LMS (though it could).

## 5.5 How to apply it

### 5.5.1 Assessment and SCOs

To decide whether to use one or multiple SCOs for an assessment, consider that no matter which one you choose, each assessment SCO can report a completion status and a result. For more details about SCOs in content design in general, see [Sharable content objects \(SCOs\)](#).

### 5.5.2 Single SCO

A single SCO assessment contains all questions in the same SCO. When the user has completed the assessment, a single rollup score and completion status will be sent to the LMS along with any individual test item information (see [Tracking Learner Data](#)) you have specified. Using one SCO, you can present one test item per screen or all test items on a single screen. You can also randomize the test items. All of these design elements, as well as the use of a test bank if desired, are handled by programming inside the SCO.

Remediation, if desired, will be based on performance on the entire test or on individual or sets of responses and is handled by sequencing that your programmer takes care of by creating rules (see [Creating adaptive content using sequencing rules](#)).

The testing experience for each learner may be better using a single SCO than if you use multiple SCOs because the SCO only communicates with the LMS once to report the final score and performance data for all questions, limiting data transfers that might introduce delays.

### 5.5.3 Multiple SCOs

A multiple-SCO assessment contains questions in separate SCOs. One form of a multiple SCO assessment has one question per SCO, with the presentation of the questions handled by the rules created by your programmer (see [SCORM Programming Terminology](#)). Another form of multiple-SCO assessment could involve the use of a test bank, where a pool of questions for each objective is contained within each SCO. In these cases, a different test question is chosen from the pool (usually randomly) and delivered to the student during each test attempt. Again, sequencing and rollup rules will govern the presentation of questions to the learner. Rollup rules consolidate the results of each SCO in the multi-SCO assessment and report a collective result at the aggregation level of the assessment.

When a test uses multiple SCOs, the LMS can receive a score and a completion status for each test item. It can also collect detailed response information for each test item, allowing the delivery of feedback and remediation based on that information. Reporting can be done at a granular level.

When would you use multiple SCOs for an assessment? One example is when you want to treat each test question separately for scoring, remediation or feedback. A second example would be for reuse. If you wish to have each question, or set of questions available for reuse within other assessments, it may be better to have them as separate SCOs.

Using multiple SCOs in an assessment requires the ISD to provide specific direction to the programmers about the ordering. The ISD will also determine how to compute the overall score for the assessment. The programmers will use “rollup” rules that facilitate programming the overall score, and “sequencing objectives” to facilitate remediation.

The primary drawback to the use of a multiple SCO assessment is the potential effect on the user experience. Since the SCO must communicate with the LMS after each question, the learner may encounter delays and disruptions during the assessment. Some LMSs may require each SCO to be separately sent to the LMS. Learners may also have to navigate between the LMS and each SCO. Finally, the multiple-SCO approach increases the complexity and thus the level of effort for development and maintenance of your content.

## 5.5.4 Scoring

You can set a passing score that determines if the learner passes or fails the assessment (or question, if using multiple SCOs) when a certain threshold is achieved or not achieved.

## 5.5.5 What LMS data can be used?

The ISD can choose to use performance data about assessments. To select data, consider the following questions:

- What kind of data is needed? Are there specific team or client requirements?
- Who will have access to the data? What security considerations are there for the data if it is reported individually?
- How will the data be used?
- Will the data be used to improve the content and the assessment?
- Will the data be used to identify individual learners who excel or perform poorly?
- What stage of the evaluation process are you in, formative or summative?
- What are the contractual reporting requirements for the project in this evaluation stage?
- How will the reported data be delivered—as aggregated or individual data?

- What type of reporting is provided by your LMS to facilitate data reporting?
- How many report options are provided?
- Are the reports easy to run and automatically formatted?

A test item can provide the following information at the SCO level (your programmer will recognize these as *cmi.interactions*):

- Order of the responses as they were presented to the learner—if the order of the responses was randomized
- Correct answer(s)—what the learner should have answered
- Learner’s response—what the learner actually answered
- Whether the learner’s response was correct or not
- Latency—duration taken to respond
- Weight of the particular item relative to the overall assessment score

**Best Practice:** The ISD should determine exactly what information needs to be used, and that can be reported by the LMS, and convey those needs to the programmer. Be aware that collecting too much data about assessments:

☞ Increases programming time (may increase cost of the project).

☞ Increases the amount of communication between the SCO and the LMS (may slow LMS response time).

☞ Increases the storage requirements of the LMS (may reduce amount of total storage available). This may be significant for LMS deployments involving large numbers of users.

☞ May violate the rights of learners in some cases (particularly in situations where learners are union employees).

## 5.5.6 Linking Assessments to Learning Objectives

ISDs use learning objectives to drive the design of course content modules, which are in turn measured by assessments. SCORM does not change that. The decisions to be made relate to whether each assessment SCO covers one or more learning objectives, what type of reporting is needed, and what remediation will be included.

## 5.5.7 What's New in SCORM 2004

The sequencing rules new in SCORM 2004 are what allow the remediation described in this section to take place.

## Chapter 6. Reusable Content

### 6.1 Introduction

The reusability of content is one of the high-level requirements of SCORM, as stated by the ADL Initiative (see [Overview of SCORM](#)). This means that content designed for one organization can be redeployed, rearranged, repurposed, or rewritten by other organizations that have similar learning needs. This section will offer guidance and best practices for the ISD to both create reusable content as well as to find and reuse existing content.

### 6.2 Example

You may have a course, like the one described in earlier sections (see [Assets, SCOs, & Aggregations](#)), about basic car maintenance (see Figure 6-1). This course is composed of several topics: how to change a flat, how to change the oil, how to maintain the brakes.

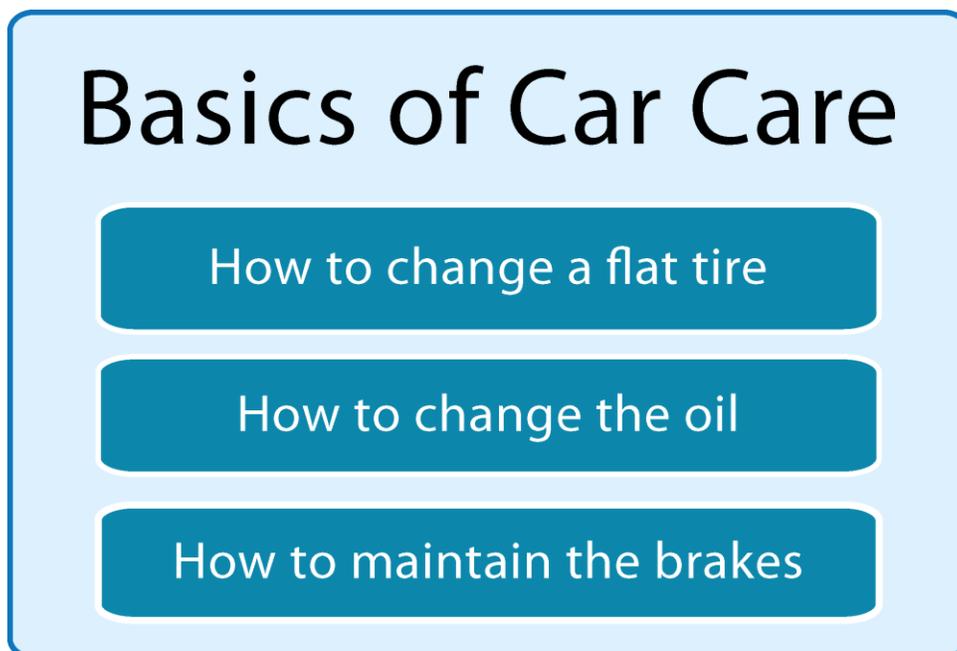


Figure 6-1 Basics of Car Care

Content of the course written in a reusable way is probably general enough to be used as is by several different organizations – perhaps a driver’s education course, a car dealership, or auto club. The content could also be reused as part of a larger course offering, as we have seen in previous sections (see [Flowcharting your sequencing plan](#)), about the things a person needs to know about car ownership (see Figure 6-2).

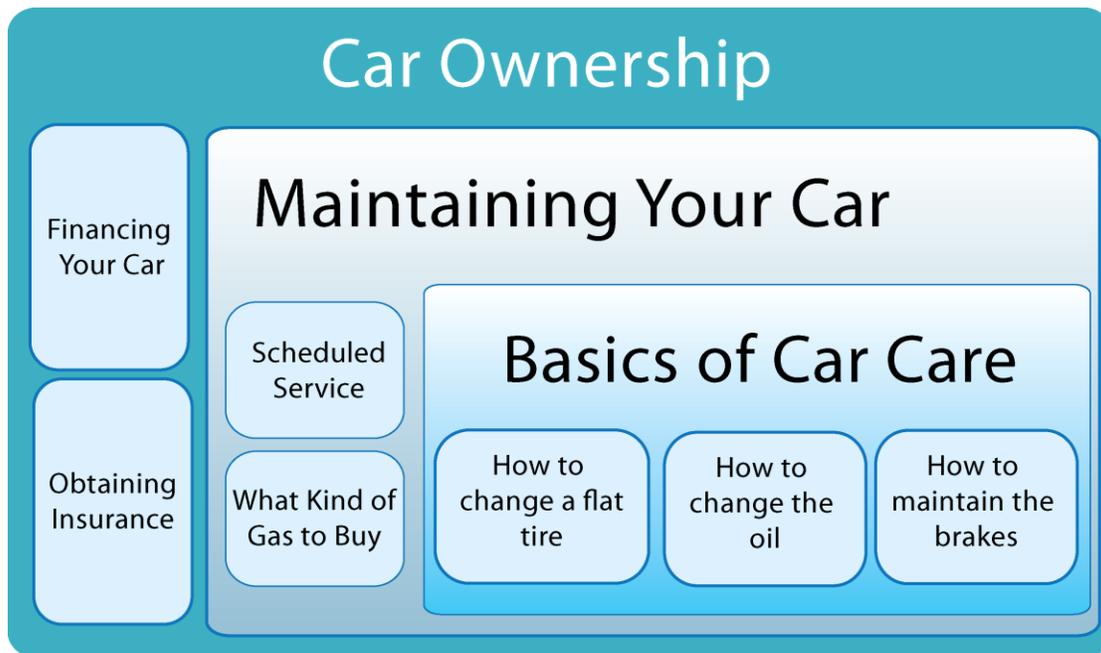


Figure 6-2 Reuse of Basics of Car Care Course in a Car Ownership Course

## 6.3 When to apply it

Any time you are creating content, you should take reuse into consideration. First, see if there is existing content that can be reused in one of the ways discussed below. Then, consider the reusability of your own content as you are designing and implementing it.

## 6.4 What SCORM does not do

SCORM does not dictate the ways in which content should be reused. SCORM also does not require that all content be reused. What SCORM does is enable the reuse of content by defining standards for how content is structured and packaged.

In terms of reuse, it is important to understand that you cannot easily give someone a SCO and have them plug it in as they would a PowerPoint slide into a slide deck. It is not a file or folder. It is part of a content package that has to be recoded by a programmer in order to reuse SCOs from it, or accept SCOs into it. Content packages are the standard way to organize and use SCORM courses. It is a .zip file containing all files (e.g., SCORM code and assets) necessary to deliver the course. For more information, consult the SCORM Users Guide for Programmers.

## 6.5 How to apply it

### 6.5.1 Defining reusable content

For as long as instructional designers and learning theorists have been talking about learning objects and reuse, they have also been debating the relevance, applicability, and utility of learning objects and reuse. When people hear the term reuse, they immediately equate it with the direct, un-changed use of what is typically highly contextualized content designed for a specific group of individual learners. This is only one example of reuse. Content can be reused in multiple ways: redeployed, rearranged, repurposed, or rewritten.

#### *Redeploying content*

Redeploying content is running the same content, with no modifications, in multiple Learning Management Systems (LMSs). When content is packaged according to SCORM standards, you can use it in any SCORM 2004-conformant system as it was designed and developed, without modification.

In the example above, the reuse of the Basics of Car Care course in a driver's education course, or by a car dealership or auto club would be considered redeployment. You would simply be importing the content package onto a different LMS. Be aware that some of the metadata may then be misleading, since it reflected the old context, so you may want to change it.

#### *Rearranging content*

Rearranging content is taking existing content and re-ordering it for new uses or contexts. Taking the Basics of Car Care course and making it a part of the Car Ownership course (in Figure 6-2) would be a rearrangement of content. The three topics in the Basics course could also be made into separate smaller courses. For example, an auto club might only want to teach how to change a flat tire, and only that SCO might be used as a new course offering.

#### **Best Practices:**

- 🔧 Small granularity of SCOs makes reuse possible.
- 🔧 Document your design so others can understand it, and keep the design as simple as possible.
- 🔧 Build content once and share it across courses in a curriculum.
- 🔧 Promote reusability within your enterprise by enforcing use of templates and style guides across the enterprise.

In summary, when rearranging content, you are either changing the order of SCOs, deleting SCOs, or both. You are not adding SCOs that weren't there to begin with.

### ***Repurposing content***

Repurposing content enables you to take the same content and use it in new contexts or in different ways. In this case you can add SCOs that weren't in the original content. For example, the How to Change a Flat Tire portion of Basics of Car Care could be repurposed and used in a course on Handling Highway Emergencies, along with other topics, as shown in Figure 6-3.

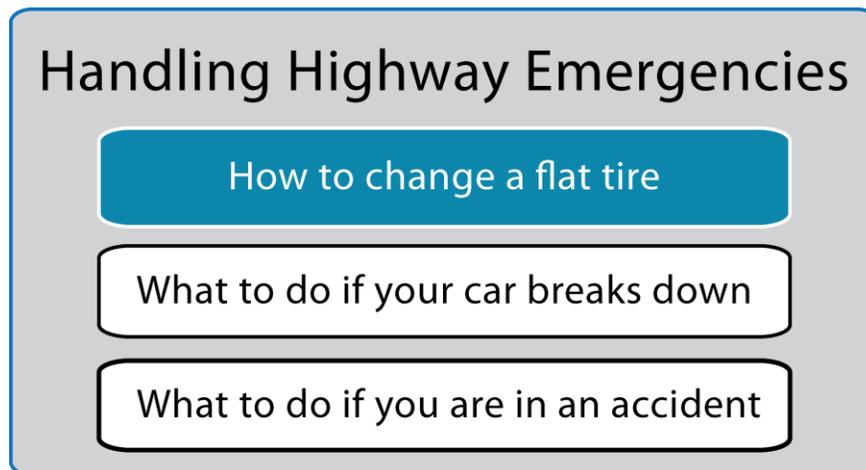


Figure 6-3 Repurposing Part of the Basics of Car Care Course

### ***Rewriting content***

A significant portion of the time you spend designing new content is devoted to researching and authoring the content. If the content is designed so that it is highly granular, and therefore more easily reused, then taking the relevant materials and changing the examples or imagery, rewriting the verb tenses, modifying the person or voice of the subject for different audiences, or removing irrelevant information can save you effort that you can then apply to creating a more valuable learning experience.

In the rewriting paradigm, you are making changes to assets in SCOs (per examples described above) and not using them as-is. This involves more than just making changes to the manifest file, as in the case of repurposing and rearranging. Merely rewriting content is technically outside the scope of SCORM, since you are simply updating content information and not changing SCORM parameters.

For example, the Basics of Car Care course might be rewritten to address specific models or types of vehicles. Using the general content as the base, the model-specific content could be added in as needed to produce a new course.

## 6.5.2 Creating reusable content

### *Granular size*

The first step in creating reusable content is creating granular content. This means that SCOs should be the smallest size possible for your purposes. A piece of content that addresses a broad subject like Maintaining Your Car might be too comprehensive to be reusable for a variety of audiences. However, a smaller piece of content, such as How to Change a Flat Tire, will be more reusable because the procedures involved are fairly consistent and unrelated to a particular type of vehicle, as well as applicable to a variety of different learning situations.

Although smaller SCOs are better in most cases, there may be a point of diminishing returns. Increasing the number of SCOs does increase the complexity and level of effort to code the manifest file and manage the SCORM elements. If you are dealing with a large course consisting of hundreds of screens and many hours of seat time, you may not want every topic to be a SCO; it may be better to consolidate some of them or use the next level of instructional organization (for example, lessons) to map to SCOs. You may even want to think about making each module of the course a separate SCORM organization.

To determine how best to divide or chunk your content, look at the potential audiences you have identified and ask yourself the following questions:

- What portions of this content apply only to my target audience?
- Can the content be divided before and after the portion that applies only to my target audience?

As mentioned earlier, it is generally a good idea to consider mapping content units associated with learning objectives to SCOs. You need to determine how this fits with the above considerations.

### **Context-neutral content**

The other key to keeping content reusable is making it context-neutral. This means that a particular SCO can be separated from its package and still be considered complete—that is, understandable to the learner without the need to refer to critical information that is not in the SCO. When you limit a SCO to a single granular topic, perhaps even a single learning objective, it is easier to make it context-neutral.

Where context-specific instruction is required, you can create context-specific SCOs to address those objectives. For example, if you wanted to reuse the Basics of Car Care course for a specific type of vehicle, the best way would be to use generic SCOs, like Changing a Flat Tire, and interweaving new SCOs that address the specific vehicle you need to include.

#### **Best Practices:**

- ☞ Don't refer to things that might change like course numbers, lesson numbers, or people's names – like "For tech support, contact Mike Smith."
- ☞ Consider the stability of Internet hyperlinks before you include them.
- ☞ Design a SCO with an asset containing a master list of all links in the course to make it easier to make global changes to URL links.
- ☞ Avoid references to previous course material and to specific features. For example, use "PHP is an effective way to create web applications" not "The Acme IT Department uses PHP to create web applications because it is so effective."
- ☞ Context across SCOs can be created with voice narrations, independent of the visuals.
- ☞ Don't use extraneous items (e.g., a particular corporate setting) that seem to make it apply only to that context.
- ☞ If you work at the level of html or style sheets for formatting, use CSS instead since it allows for more separation of content from appearance.

### 6.5.3 Finding reusable content

Your design process may already include researching the availability of reusable content that might be appropriate for your project. This may uncover courses, SCOs, or assets created by your own organization that you can reuse. You can also look outside your organization for content to reuse. The investment of time and effort needed is usually not significant, and will probably be worthwhile if you find content that saves you design and development time.

It is important that you check and respect Intellectual Property (IP) rights of any content that you wish to reuse. Some content will be tagged with metadata describing the IP restrictions and requirements; for others, you will have to contact the content owner to determine this, and possibly obtain written permission from them.

See the [Glossary of SCORM Terms](#) for a list of other sources of reusable content.

#### Best Practices:

- 🏆 Search for reusable SCOs instead of creating all of them from scratch! This may be a new paradigm for you, but it is a habit that can reap tremendous savings for you and your clients.
- 🏆 Update and adapt resources you previously created.
- 🏆 Use social media like Facebook or LinkedIn. Google can be great, but it doesn't delineate learning content from books or other kinds of media.
- 🏆 Contact other people to see if they have something you could use.
- 🏆 Make your material available to others.

### 6.5.4 Metadata

Metadata is literally data about data. It's not a SCORM-specific concept. You can think of metadata as similar to the information that used to be found in a card catalog at a library. You may encounter metadata frequently online—it can be key words, subjects, topics, authors, etc.

**Best Practice:** Think about web-based ecommerce when considering metadata tags. Tag your content carefully if you want people to find it.

In SCORM, metadata is the information that describes your content, both the individual components (the assets and SCOs) and the content packages. In SCORM packages that you and your programmer create, the metadata will be contained in the manifest file (see [Glossary of SCORM Terms](#) for definition). SCORM has a specified structure for creating the manifest file, but the actual metadata that is included can vary. Your organization may have its own requirements for metadata, depending on how it will be used.

There are various editors (including the open-source RELOAD editor offered by the ADL Initiative) to help you enter the metadata into the manifest files. These editors generally provide an easy-to-use interface to allow you to enter the appropriate information in the

fields you wish to use. They will then output an appropriately formatted file which can then be incorporated into the manifest file by your programmer. There are many open source and commercial editors available; some can automatically extract or infer some metadata by analyzing the content, saving you time.

**Best Practice:** Choosing modules off a shelf will probably not meet your learners' needs. You'll want more information about how it was used originally and what type of analysis or assessment was done to identify those needs. Think about this when you choose your own metadata tags, too.

## 6.6 What's New in SCORM 2004

Metadata tags changed somewhat from SCORM 1.2 to SCORM 2004. These changes were made primarily to incorporate industry standards into the SCORM model. Metadata records use the Institute of Electrical & Electronics Engineers (IEEE) Learning Object Metadata (LOM) format, which is a standard, internationally recognized model for storing and retrieving metadata about objects used in learning applications. Your programmer is likely aware of these changes. If you are responsible for creating the metadata and manifest files for your SCORM packages, make sure you are using the most up-to-date information from the ADL Initiative. If you use a tool to do this, make sure it supports SCORM 2004.

## Chapter 7. Tracking Learner Data

### 7.1 Introduction

SCORM data model elements (DMEs) allow the LMS to collect and store data about learners and their progress through the SCO. After the SCO has initiated communication with the LMS, the SCO can store information in the LMS or ask the LMS to report previously stored values. A SCORM 2004-certified LMS is required to support all of the SCORM data model elements, but you are not required to use any of them. As an ISD, you need to know what data can be communicated through SCORM data so you can tell your programmers what you want to happen.

### 7.2 Example

You might want to customize your learners' experience in your course in one of these ways:

- Refer to them by name inside the content (i.e., “Welcome to the Car Care Basics Course, Jane.” “Well done, Mr. Smith.”; DME `cmi.learner_name`)
- Remember the last location in the content the learner viewed (i.e., “Do you want to start where you left off?”; DME `cmi.location`)
- Use preferences of the learner (i.e., use audio transcript, language choice, etc.; DME `cmi.learner_preference`)

You may also want to store information in the LMS about the learner’s experience with the SCO:

- How they performed in the SCO (e.g., `cmi.completion_status`)
- How much total time they spent in the SCO (e.g., `cmi.total_time`)
- How much time they spent in a certain session on the SCO (e.g., `cmi.session_time`)
- Whether or not they completed the SCO (e.g., `cmi.completion_status`)
- How they answered assessment items (e.g., `cmi.interactions`)
- If they passed or failed the SCO (e.g., `cmi.success_status`)

### 7.3 When to apply it

Any time you wish to have communication between your SCO and the LMS, you should use the SCORM data model elements. Many LMSs may be capable of storing this kind of information using native programming; however, if you use LMS-specific programming, if your SCO is reused in a different LMS, the data may not work in the same way. SCORM data

Tracking Learner Data

gives you a way to track this information, regardless of how and where your content is being used (as long as it's a SCORM-conformant LMS).

## 7.4 What SCORM does not do

SCORM does not dictate what kind of information you must collect, or how it must be used. SCORM simply gives you the ability to track and access certain kinds of information about your SCO.

## 7.5 How to apply it

Typically, the ISD is not responsible for knowing how to set up the SCORM data model elements. Your programmer will take care of that for you. What the ISD does need to establish, and be able to communicate, however, is what kind of data needs to be collected, and how that data will be used. Table 7-1 lists all of the data model elements, and how you may be able to use them. The technical (cmi) name is provided so you can be familiar with the terms when discussing with your programmer.

Table 7-1 SCORM Data Model Elements

Data Model Element with <i>cmi</i> Name	Description	Application
<b>Technical Data</b>		
Credit cmi.credit	Indicates if the learner will receive credit for performance in the SCO.	Learners can take a SCO for credit or no credit. No credit simply means that the LMS is not tracking performance in the SCO.  The default value is that the learner will receive credit.
Entry cmi.entry	Indicates whether the learner has previously accessed the SCO. If they have, the LMS will know that data for the SCO exists already.	When learners enter a SCO for the first time, the element is set to ab-initio. If the learner is re-entering a suspended session, the element is set to resume.
Launch Data cmi.launch_data	Provides data specific to a SCO that the SCO can use for initialization.	Allows SCOs to be configured with data from the LMS at the time the SCO is launched. For example, importing new statistical data into a SCO on statistics.
Location cmi.location	Represents a location in the SCO.	Used for book-marking the learner's position in a SCO in a given instance, allowing the learner to resume the SCO at the same point at which learning was suspended.

Tracking Learner Data

Mode cmi.mode	Identifies how the SCO may be presented to the learner.	Browse is typically used when learners do not want information about their performance to be stored.  Normal is used when learners want to store information about their performance.  Review is used by learners who have already completed the SCO but want to view the item again as a refresher.
Suspend Data cmi.suspend_data	Provides additional space to store and retrieve data between learner sessions.	If learners start the SCO, but do not complete it, the state of suspension (where the learner stopped) is tracked. Other information can also be stored here such as previous answers that may impact what happens in the rest of the SCO.
<b>Content Initialization</b>		
Completion Threshold cmi.completion_threshold	Identifies a value for comparison with the learner's progress in a SCO to determine if the SCO should be considered complete.	The completion threshold, determined by the ISD, allows a SCO to set its status when something happens (i.e., a specific number of pages have been viewed).
Learner ID cmi.learner_id	Identifies the learner on behalf of whom the SCO was launched.	This is a unique alpha-numeric code that refers to a single user. It allows the LMS to associate data with a specific learner.
Learner Name cmi.learner_name	Allows the SCO to present the name of the learner inside the content.	Typically used to customize learning content, for example:  "Welcome back Sgt Thomas." or "Nice work, Capt. Brown."
Learner Preference cmi.learner_preference	Specifies preferences associated with the learner's use of the SCO.	Can be used to present learner-specific content within a single SCO. If there are multiple languages in a single SCO, this data model element would select the learner's preferred learning language.  If the SCO is Section 508-compliant, it can select the audio level, delivery speed, or audio captioning.
Maximum Time Allowed cmi.max_time_allowed	Indicates the amount of accumulated time the learner is allowed to use a SCO in an instance.	The learner attempt begins with the beginning of the first session and continues until the activity terminates.
Time Limit Action cmi.time_limit_action	Indicates what the SCO should do when the maximum time allowed is exceeded.	If the learner exceeds the prescribed time limit, the ISD can choose to either automatically exit the SCO or allow the learner to continue. A standard message such as, "Your session has ended because you have exceeded the time limit" may also be included.

Tracking Learner Data

Score Reporting		
Completion Status cmi.completion_status	Indicates if the learner has completed the SCO.	The completion status, determined by the ISD, can be a test score, navigation through content, completion activities, etc.
Interactions cmi.interactions	Describes a collection of learner responses, such as responses to questions or tasks for the purpose of measurement or assessment.	Frequently used in tests or quizzes to collect learner response information. Refer to <i>Section 6—Using Assessments in SCORM Content</i> , for more information on <i>cmi.interactions</i> .
Objectives cmi.objectives	Specifies learning or performance objectives associated with a SCO.	May be used to impact sequencing decisions or how you divide your content.
Progress Measure cmi.progress_measure	Identifies how much progress the learner has made toward completing the SCO.	The progress can be based on the completion of a certain number of objectives related to the SCO, the number of pages presented to learners, etc.
Scaled Passing Score cmi.scaled_passing_score	Identifies the scaled passing score required to master the SCO.	Should be set to your minimum passing score.
Score cmi.score	Identifies the learner's score for the SCO. A SCO can only report one score.	This is typically the result of some interaction the learner has with the content (for example, answering questions on a quiz, completing a task, etc.).
Success Status cmi.success_status	Indicates if the learner has mastered the SCO.	The criteria, defined by the ISD, can be based on a percentage of interactions being passed or objectives being met, a score for a test or quiz, etc.
Comments		
Comments From Learner cmi.comments_from_learner	Enables the collection of comment text, the location where the comment was made, and the timestamp of when the comment was made.	Learners can provide text feedback (comments, suggestions) about the SCO or their learning experience, while in the content, that can later be used to update or improve the learning experience.
Comments From LMS cmi.comments_from_lms	Contains comments and annotations intended to be made available to all learners.	Instructors can provide information intended for all learners viewing the content, such as a clarification of an issue or information between updates to the content.
Exit Data		
Exit cmi.exit	Indicates how or why the learner left the SCO.	Can be used to impact sequencing decisions.
Session Time cmi.session_time	Identifies the amount of time that the learner has spent in the current instance of the SCO.	The ISD defines the value and meaning of session time. The LMS uses this time to compute total time.

## Tracking Learner Data

---

Total Time cmi.total_time	States the learner's cumulative time for all sessions of a specific SCO for a given learner attempt.	Stores the total time spent in every session of a given SCO, for a given learner.
------------------------------	--	---

---

## 7.6 What's New in SCORM 2004

SCORM 2004 raised the character limit for interactions data from what was allowed in SCORM 1.2. This means that more data can be stored by the LMS.

# Appendix: Sources of Reusable Content

Other than using personal networking to find content, here are some possible sources:

**ADL 3D Repository** (<http://3dr.adlnet.gov/>) is a website for uploading, finding, and downloading 3D models. Each model stored there has an associated license that is selected when the model is uploaded. It currently includes Creative Commons licenses and public domain options.

**DefenseImagery.mil** (<http://www.defenseimagery.mil/index.html>) is a source of free unclassified U.S. military-related still images, motion imagery, audio/visual (A/V) and interactive multimedia instruction (IMI). The latter is another term for e-learning.

**Defense Video and Imagery Distribution System (DVIDS)** (<http://www.dvidshub.net/>) serves as a turnkey operation that facilitates requests for DoD Public Affairs video, audio, still imagery and print products.

**Google** can be useful for finding content, but it doesn't differentiate learning content from books or other kinds of media. Using terms like "e-learning" in the search terms may help filter for learning content.

**Khan Academy** (<http://www.khanacademy.org/>) is a non-profit organization that makes available videos, exercises, and assessments for K-12 subjects. To reuse you would need to link out to the asset, unless you can make arrangements with the owner of the asset to get a copy.

**Learning Registry** (<http://www.learningregistry.org/>) is a federally funded project currently under development. It will make federal learning resources easier to find, easier to access and easier to integrate into learning environments wherever they are stored, around the country and the world.

**Multimedia Educational Resource for Learning and Online Teaching (MERLOT)** (<http://www.merlot.org/merlot/index.htm>) is a free and open online community of resources designed primarily for faculty, staff and students of higher education from around the world to share their learning materials and pedagogy.

**USA.gov** (<http://www.usa.gov/Topics/Graphics.shtml>) contains photos and images by topic, related to the U.S. Government. Some of these photos and images are in the public domain and may be used and reproduced without permission or fee. However, some photos and images are protected by license.

**YouTube** (<http://www.youtube.com>) contains millions of videos on countless subjects (including some by the ADL Initiative). To reuse you would need to link to the video, unless you can make arrangements with the owner to get a copy.

# Glossary of SCORM Terms

**Accessible:** Accessible content can be loaded and accessed when needed to meet training and education requirements.

**Adaptable:** Adaptable content can be customized for individual learners and organizations as needed.

**Advanced Distributed Learning (ADL) Initiative:** An evolving, outcomes-focused approach to education, training, and performance support that blends standards-based distributed learning models, emphasizing reusable content objects, content and learning management systems, performance support systems/devices, web applications services, and connectivity.

**Aggregation:** Aggregations are used to group related content so that it can be delivered to learners in the manner you prescribe. Sequencing rules allow you to prescribe the behaviors and functionality of the content within the aggregation as well as how the aggregation relates to other SCOs within the same root aggregation.

**Application Programming Interface (API):** The SCORM API is a standardized method for a sharable content object (SCO) to communicate with the learning management system (LMS) when a learner is interacting with a SCO. There is a specific set of information the SCO can set or retrieve. For example, it can retrieve information, such as a student name, or a set of values, such as a score.

**Asset:** Assets are electronic representations of media, text, images, sounds, web pages and other pieces of data that can be delivered to a Web client. They do not communicate with the LMS directly. Assets, like the sharable content objects (SCOs) in which they appear, are highly reusable. In order to be reused, assets are described using metadata so that they are both searchable and discoverable in online content repositories.

**Bookmarking:** SCORM offers an interoperable way to save the last location the learners visited in the content. The bookmark is saved in the `cmi.location` variable, one of the interaction data elements listed in the data model section.

**Certification or Certified:** "Certification" indicates that materials have been tested by an independent third party to assess conformance with the guidelines established in SCORM. "Certification" indicates a successful testing by the Conformance Test Suite. All "certified" products are "conformant."

**Compliance or Compliant:** A product is compliant when tested to ensure it performs according to applicable guidelines, instructions, policy, or law. The SCORM test suite is designed to rigorously test inputs, processes, and outputs.

**Conformance or Conformant:** A product or service is conformant when it adheres to technical specifications, guidelines, recommendations, or best practices to identify the correctness, completeness, and quality of developed product or service. Test assertions are achieved by

inspecting results focused on reliability, stability, portability, maintainability, and usability. No form of testing is used other than evaluating actual results against expected results.

**Content Package:** The content package contains everything needed to deliver the course, module, lesson, etc. to the learner via the LMS. A SCORM content package contains two principal entities: (1) a manifest file that lists all of the resources or assets you want to include in the package, the content structure you created (called the organization), the sequencing rules, and all of the metadata for the SCOs, the LOs, and the package itself; (2) all of the physical SCO and asset files for the content.

**Context-neutral Content:** Context-neutral content can be separated from its SCORM package and still be considered complete. You can maintain context-neutrality by not referring to other SCOs, avoiding direct links to other SCOs or portions of the content, etc.

**Data Model Elements:** A set of information about a learner's performance in, and interaction with, the instructional content initiated by the SCO and stored in an LMS. Data model elements are made interoperable by the SCORM Run Time Environment Data Model. The SCORM data model elements allow the LMS to collect data on the learners and their progress through the SCO. This data can then be used by the LMS for reporting purposes and to offer personalized content.

**Durable:** Durable content does not require modification to operate as versions of software systems and platforms are changed or upgraded.

**Interactions:** Interactions are the items of information about a learner's responses on an assessment. This information is sent to and stored by the LMS.

**Interoperable:** Interoperable content operates across a wide variety of hardware, software, operating systems, and web browsers regardless of the tools used to create it and the platform, such as a learning management system (LMS), on which it is initially delivered. SCORM-conformant content can easily be moved from one SCORM-conformant LMS to another.

**Learning Management System (LMS):** An LMS is a software package used to administer one or more courses to one or more learners. An LMS is typically a web-based system that allows learners to authenticate themselves, register for courses, complete courses and take assessments. The LMS stores the learner's performance records and can provide assessment information to instructors.

**Manifest:** A manifest is a description of everything contained in your content package. Generally a tool such as the Reload Editor will be used to create the manifest as an XML document during the content packaging process. The manifest includes all of the resources or assets included in the content package, the content structure you created, the sequencing rules, and all the metadata for the SCOs and the package itself.

**Metadata:** Metadata is "data about data." It is the information that describes what your content is, both the individual pieces (the assets and SCOs) and the content packages. Metadata enables instructional designers searching for content or assets to locate them with relative ease and determine whether they will be useful before downloading or requesting rights to your content.

**Objective:** In traditional instructional design, a learning objective is used to measure the attainment of a knowledge, skill, or ability in accordance with a predefined behavior, a prescribed condition, and an achievement standard. Your SCOs may each contain one objective or several objectives – it’s up to you.

Your programmer may also use the term objectives, but it does not necessarily mean the same thing. When the programmer sets up sequencing, which are the rules that guide the way the content is presented to the learner, the variables that can store information about one SCO in the LMS that can be retrieved for later use to impact another SCO are also called "objectives." The programmer can use these objectives, at the ISD’s direction, to do remediation automatically.

**Organization:** The organization is the part of a content package where SCOs are ordered into a structure and sequencing behaviors are assigned to them. The organization outlines the entire structure you have created for your content. The organization provides order to the otherwise unordered collection of SCOs and their metadata.

**Remediation:** Remediation is used to help learners comprehend instruction with which they may be struggling. In the event that learners do not answer a test item correctly, or fail an entire test, they can remediate to already-viewed content for review or completely new content to try and understand from a different approach. In SCORM 2004, remediation paths between SCOs can be created using sequencing rules.

**Repository:** A repository is a device for storing and maintaining digital information (content).

**Reuse:** Reused content is existing content used in new or different contexts or applications. SCORM content can be reused in multiple ways: redeployed, repurposed, rearranged, or rewritten.

**Root Aggregation (see also Organization):** A root aggregation is a top-level aggregation.

**Sharable Content Object (SCO):** In general terms, a SCO is a collection of assets that becomes an independent, defined piece of instructional material. SCOs are the smallest logical unit of instruction you can deliver and track via a learning management system (LMS).

**Sharable Content Object Reference Model (SCORM):** SCORM is a model that references and integrates a set of interrelated technical standards, specifications, and guidelines designed to meet high-level requirements for e-learning content and systems.

**Sequencing:** Sequencing is similar to the ISD term “branching” in that it describes and prescribes the manner in which learners receive content. In SCORM, the “branching” is directed by the sequencing rules your programmer creates, based on your design. Based on these rules, the learning management system (LMS) sequences all the sharable content objects (SCOs).

# Index

adaptive content.....	10, 12, 25	metadata.....	44
aggregation.....	9	modules.....	9
assessments.....	32	navigation .....	19
asset.....	9	objective .....	54
bookmarking.....	30	organization .....	9
branching .....	19	outline.....	12, 29
check on knowledge.....	34	personalization .....	12, 30
content design .....	27	posttest.....	32
context.....	13, 43	remediation .....	37
course .....	9	reusability .....	28
customize .....	46	reuse.....	13, 28, 42, 44
decision points .....	12, 29	SCO size.....	13, 29, 42
flowchart .....	21	scoring.....	34
knowledge checks.....	34	SCOs.....	9, 28
learning objective.....	54	sequencing.....	21
LMS .....	21	sharable content objects .....	9, 28, 32, 34
LMS data .....	12, 30, 46	storyboards .....	29