



Using SCORM-Compliant Content With Saba

Version 2.0

Table of Contents

About This Guide	4
Intended Audience	4
Terminology.....	4
Courseware Development Process.....	6
Courseware Development Lifecycle Overview.....	6
SCORM Essentials.....	12
SCORM Versions	13
SCORM 1.2.....	13
SCORM 2004	13
Comparison Between SCORM 1.2 and SCORM 2004	14
SCORM 2004 Support in Saba	15
Communicating With Content	16
Communicating With Content in Saba	16
Content Storage Setup.....	18
Configuration	20
Global Settings.....	20
Player Templates	22
System-Defined Templates.....	22
Empty Player Templates.....	22
3.x or 5.1 Compatible Player Template	23
Player Template Properties.....	23
Content Level.....	25
Is Scoring	25
Hide Exit Options (Only for SCORM 2004 Content)	26
Hide Exit Without Saving Option (Only for SCORM 2004 Content)	27
Discard Current Attempts	27
Content Management.....	29
Importing Content.....	29
Updating Content	30
Removing Content.....	30
Content Working	31
Content Launch.....	31
Content Tracking.....	32
Determination of lesson status for SCORM 1.2	32
Locking for the lesson status and score	32
Determination of completion status for SCORM 2004	33
Determination of success status for SCORM 2004	33

Support for interactions	33
What is not supported in Saba?	33
Recommendations for content vendors:.....	34
Content Sequencing	34
Content Rollup	35
Completion Logic	35
Content Completion Logic.....	35
Content Module Completion Logic.....	36
Content Rollup Failure.....	37
SCO Completion (For SCORM 2004 Contents)	37
Micro Rollup (Content Module Level).....	37
Before SCORM 2004 3 rd Edition Support	38
After SCORM 2004 3 rd Edition Support.....	39
Macro Rollup (Offering Level)	40
Concepts.....	40
Offering Level Rollup.....	40
Module Status Derivation.....	41
Problem Solving.....	43
Tools.....	43
Content Compliance Test Tool	43
HTTP Capture Tools	44
Saba Content Communication Log.....	44
Common Problems.....	45
Content Import.....	45
Content Communication	45
Server Configuration	46
Network Infrastructure	46
Frequently Asked Questions (FAQs).....	47
References.....	49

About This Guide

This document provides an insight into the Saba application's implementation of the SCORM specification for managing and tracking Web-based learning objects, and discusses the elements of the SCORM specification that are implemented in Saba's learning solution.

Intended Audience

The following table describes the intended audience types for this document.

Table 1: Intended Audience

Audience Type	Document's Intended Use
Saba's Learning Solution Users	This document describes how to load and launch SCORM-compliant content in the Saba application.
Saba's Content Providing Partners	This document provides guidance about how to use the Saba application to develop SCORM-compliant interfaces within their WBT courses. These interfaces between the Saba application and the SCORM-compliant content enable courses, learning objects, and course players to interoperate and exchange data with the Saba application's learning solution.

Terminology

The following table describes the terms that are used in this document.

Table 2: Terminology

Term	Description
SCORM	Sharable Content Object Reference Model aims to foster the creation of reusable learning content as "instructional objects" within a common technical framework for computer-based and Web-based learning.
AICC	The Aviation Industry CBT (Computer-Based-Training) Committee. It addresses the need for standardization of CBT delivery platforms within the aviation industry.
LMS	Saba's Learning Management System.
TOC	The Table Of Contents. It holds the lesson details of the content, which is used by the Player while rendering content.
SCO	Shareable Content Object
ADL	Advanced Distributed Learning

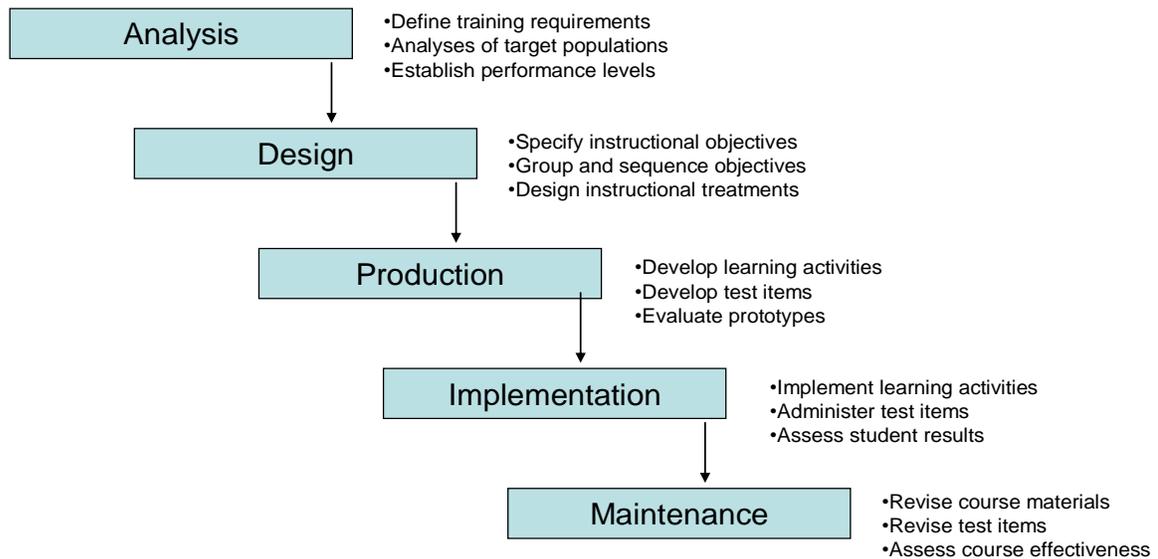
CMI	Computer Managed Instruction
-----	------------------------------

List not complete. To be extended.

Courseware Development Process

Courseware Development Lifecycle Overview

Basic Courseware Development Lifecycle



Analysis

- Define training requirements
- Analyses of target populations
- Establish performance levels

Abstract:

During the Analysis phase of courseware development the client has come to the training department with a specific need for on-line education. At this point the training department would assemble to explore and define the training needs of the target audience. The expectations for the courses are defined to effectively reach the target audience and the methods to be used to achieve the expected performance levels (Further details may be required by the Education Specialist to propose effective solutions.) are proposed. These concepts are educational philosophies that are beyond the scope of this document.

People:

- Training Administrators
- Clients requesting education
- Subject Matter Experts (SME)
- Courseware Developer (optional)
- Project Manager
- Educational Specialist

Production

- Develop learning activities
- Develop test items
- Evaluate prototypes

Abstract:

In the Production Phase the course goes from storyboard to development. The developers and programmers are involved at this point creating the course with an approved tool.

During this process testing plans should be developed to include UI testing, functionality testing, educational effectiveness (Usually prepared by the educational specialist) and Communication testing for conformance. This testing can cross over into the deployment cycle due to the range of testing needed for the course.

People:

- Training Administrators
- Subject Matter Experts (SME) (Optional)
- Courseware Developer
- Graphic Artist
- Project Manager

Implementation

- Implement learning activities
- Administer test items
- Assess student results

Abstract:

Implementation Goes from the construction to the deliverable. In this sub cycle of the development the course is tested in the various stages of delivery. This can be up to the point but not moving to production. This is a sub cycle since the course must meet the testing criteria for each test plan. Failures will force the course to go back to development to provide fixes to code, images, text, etc. It should not go back to requirements gathering.

After the success of the course in the testing it is then able to move to production. Production roll out planning should be taking place during the testing phase. It is advised that well thought out planning for the roll out will help reduce the strain on the network and application. Large deployments should be scheduled and monitored to not conflict with other large deployments.

People:

- Training Administrators
- Clients requesting education
- Subject Matter Experts (SME)
- Courseware Developer
- Project Manager
- Testing Groups
- Saba Deployment Team (Optional)

Maintenance

- Revise course materials
- Revise test items
- Assess course effectiveness

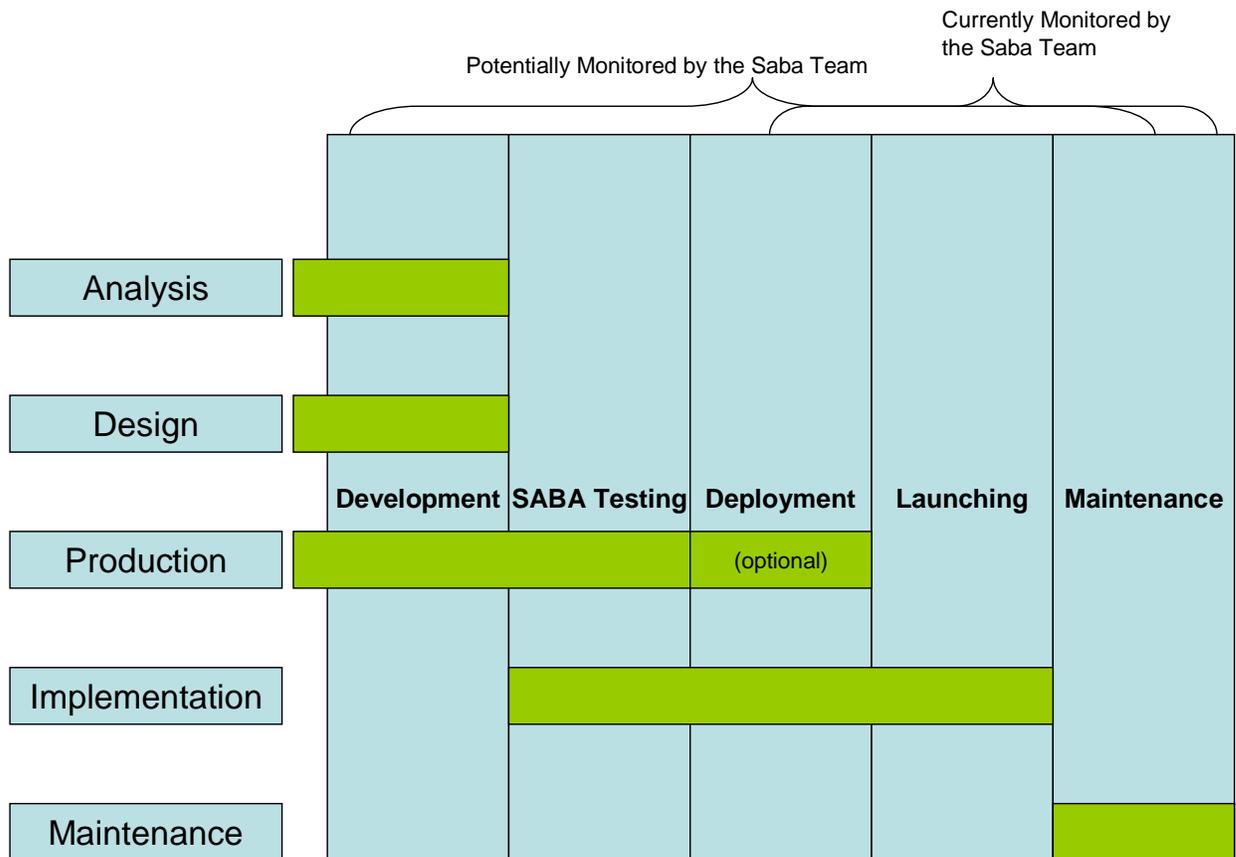
Abstract:

Maintenance of courseware development from the development perspective is only relative to revising content, assessing the effectiveness of the course, repairing any communication issues and refreshing courseware text, images, etc...

This side of maintenance has no relation to the maintenance that the SABA Team would be performing for loading the course to production and versioning the online course.

People:

- Training Administrators
- Clients requesting education
- Course Developer (If revisions are needed)
- Project Manager
- Saba Deployment Team
- Educational Specialist



SCORM Essentials

The Advanced Distributed Learning (ADL) initiative is a collaborative effort between government, industry and academia to establish a new distributed-learning environment that permits the interoperability of learning tools and course content on a global scale. Launched in 1997 by the Department of Defense and the White House Office of Science and Technology Policy, the ADL is chartered with developing an open architecture for standardizing the reusability and interoperability of online learning content.

The ADL *Shareable Content Object Reference Model (SCORM)* specification provides a common technical framework for packaging and delivering Web-based learning and fosters the creation of modular learning content in the form of reusable learning objects. SCORM is a collection of standards and specifications adapted from multiple sources to provide a comprehensive suite of e-learning capabilities that enable interoperability, accessibility and reusability of Web-based learning content.

The SCORM specification defines two critical but distinct aspects of interoperability for Web-based learning content:

- An aggregation model for packaging learning content
- An API for enabling communications between learning content and the system that launches and tracks it

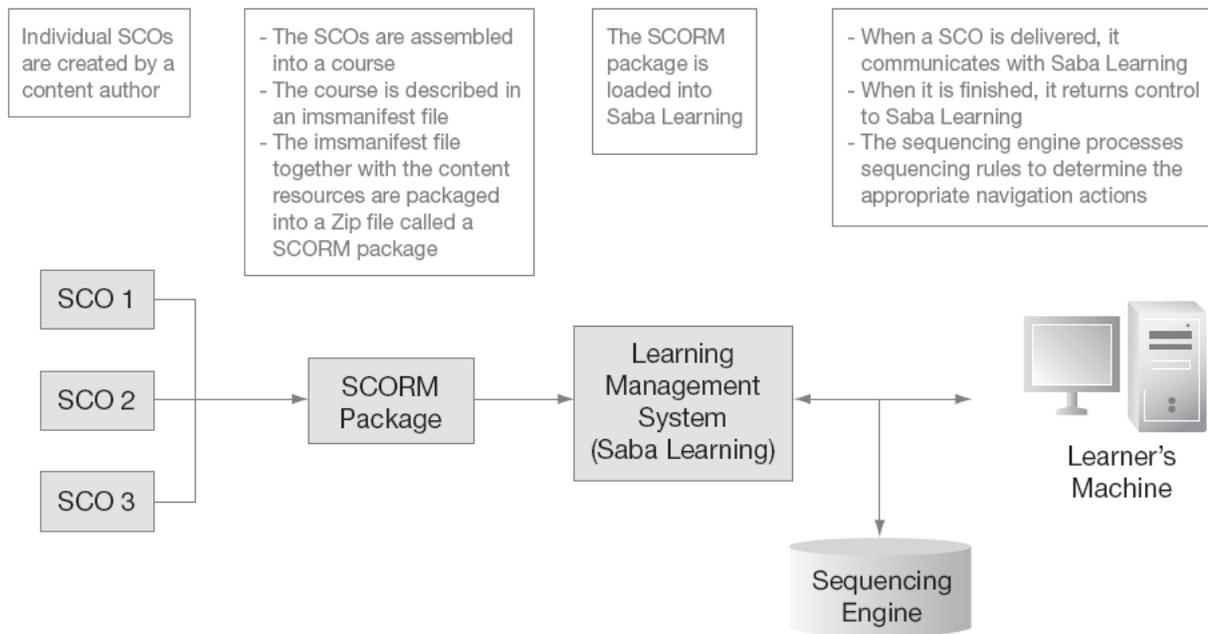
The key actors in the SCORM model are:

- Learning Management Systems (such as Saba LMS)
An LMS is any system that keeps learner information, can launch and communicate with Shareable Content Objects (SCOs), and can interpret instructions that inform which SCO to deliver next.
- Shareable Content Objects
Shareable Content Objects (SCOs) are standardized forms of reusable learning objects.

Other actors in the SCORM model are tools that create SCOs and assemble them into larger units of learning.

The following diagram illustrates the conceptual model used by SCORM to define interoperability:

Figure 1: Conceptual Model for SCORM Interoperability



Note: SCORM itself is based on other learning standards. Its content-packaging format is an application of the IMS Content Packaging specification; its runtime data model is derived from the AICC standard; its metadata model is derived from the IEEE Learning Objects Metadata standard.

For more information about the SCORM specification, see the ADL Web site at www.adlnet.gov.

SCORM Versions

Saba supports the following versions of SCORM:

- [SCORM 1.2](#)
- [SCORM 2004](#)

SCORM 1.2

SCORM 1.2 defines an aggregation model for assembling content into a portable package. The package includes a manifest file that declares the contents of the package and can optionally include the physical content resources. The packaging format defined by the SCORM Aggregation Model is based on the Content Packaging specification developed by the IMS Global Learning Consortium.

SCORM 2004

SCORM 2004 is the latest version of the SCORM specification. SCORM 2004 introduces a sequencing and navigation model that allows authors to control the way users navigate between

SCOs by defining sequencing rules that are interpreted dynamically at runtime. This allows content authors to design adaptive learning experiences where each individual user is routed through the content based on factors such as performance results, navigation choices and even user profile characteristics. For example, an author might design a learning experience that presents a pre-test followed by some learning content and one or more post-tests with logic that allows users to test out successfully by completing the pre-test or to take the learning content and successfully complete one of the post-tests. Or, an author might design a learning experience that looks at profile characteristics of the user such as location, job, language or learning style preference and then chooses the appropriate navigation path accordingly.

SCORM 2004 content is sequenced through a set of learner-initiated or system-initiated navigation events, and the branching and flow of the content can be described by a predefined set of activities, typically defined at design time by the content author. A SCORM-conformant LMS interprets the sequencing rules expressed by a content author along with the set of learner-initiated or system-initiated navigation events and their effects on the runtime environment.

SCORM 2004 represents a significant advancement in SCORM technology as it allows content authors to build SCORM packages consisting of multiple SCOs and then define navigation flows for controlling the learning experience. In previous versions of SCORM, SCOs within a single package were assumed to be fully independent and there was no mechanism for allowing the SCOs to communicate with one another. This prevented content authors from being able to exert any control over the way learners navigated the SCOs in a package. As a result, many content authors opted for the monolithic single-SCO design approach in which the entire learning experience is built into a single SCO wherein the author is able to define all the logic necessary to control navigation. The problem with the single-SCO approach is that it essentially nullifies the modularity and reusability that was intended with the SCO approach. With its sequencing and navigation model SCORM 2004 solves this problem and effectively resurrects the SCORM vision of modular reusable content.

Saba supports the following editions of SCORM 2004:

- SCORM 2004 2nd Edition
- SCORM 2004 3rd Edition

Comparison Between SCORM 1.2 and SCORM 2004

The following figure illustrates the differences between SCORM 1.2 and SCORM 2004.

Figure 2: SCORM 1.2 v/s SCORM 2004

1.2	2004
<ul style="list-style-type: none">• Proved to be portable and interoperable• Conformance has proven to lower problems with courseware and LMS interoperability• Some ambiguous standard conformity. Not all packages are created alike.	<ul style="list-style-type: none">• Sequencing Support<ul style="list-style-type: none">-Multiple Sequencing (Go anywhere, Linear, Prescript)• Strict conformance to IEEE standards. ALL resources need to be identified in the imsmanifest.• Set and Get Objective data and not just set.• Greater ability to capture reporting data at a granular level.• More scope for errors.

SCORM 2004 Support in Saba

Saba supports multiple versions of the SCORM specification, including both SCORM 2004 and SCORM 1.2. This includes the ability to import any SCORM 2004 or SCORM 1.2-conformant content, launch the content in the Web-based Saba content player, communicate with the content during the learning process through the SCORM JavaScript API, interpret and execute all sequencing actions defined by the content, and track all learning results for subsequent reporting and analysis.

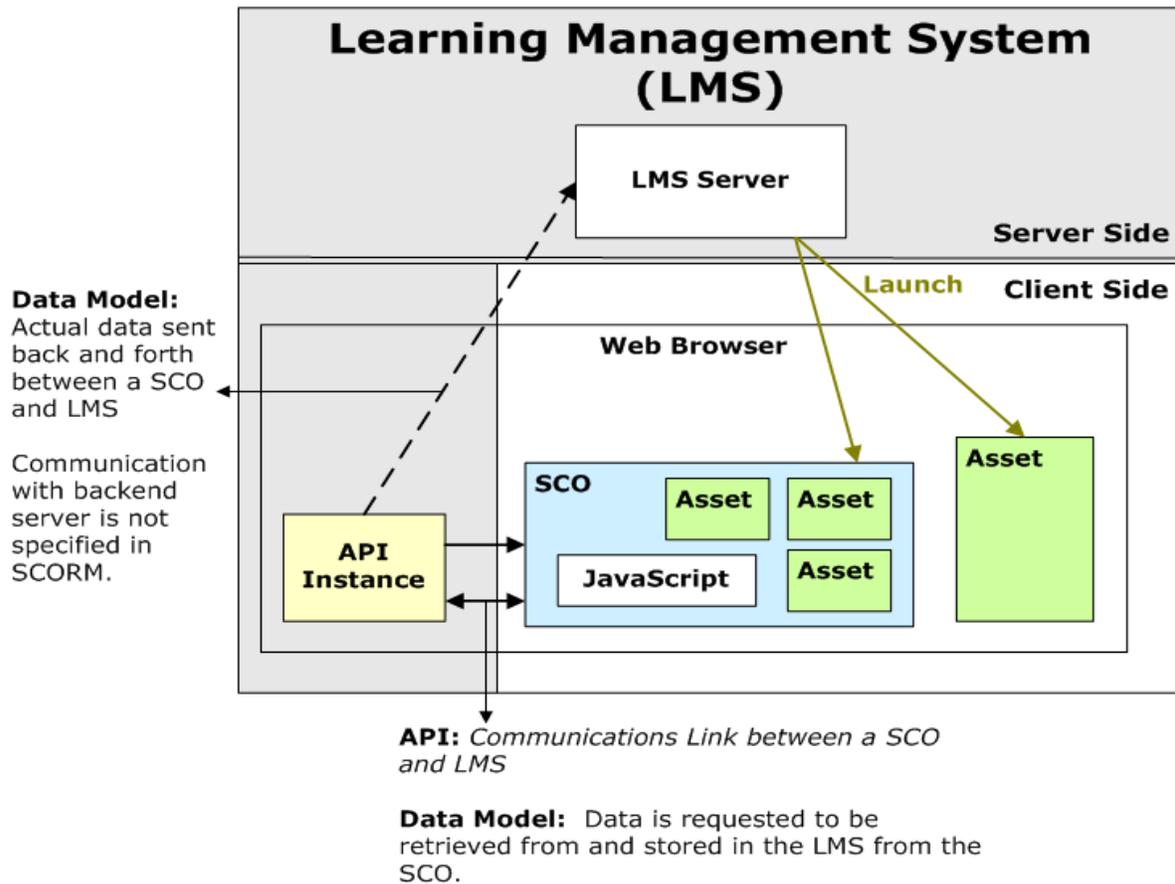
Saba supports the full SCORM runtime data model as well as the ability to parse SCORM metadata from content at import time and write it into the Saba metadata framework. Saba also provides a sequencing engine that supports the full range of sequencing behaviors that can be defined for SCORM 2004 content.

When a learner on Saba launches SCORM-compliant content, Saba passes information to the content about the learner, such as profile and preference information and information about any bookmarks and other data that might have been set during previous visits to the content. During the learning session, the Saba sequencing engine dynamically processes information about user choices and user performance, interprets sequencing rules defined by the content and sends information to the player about which SCO or SCOs can be delivered next. When the learner completes a learning session, the content passes results information back to Saba, including detailed information about learner responses to questions on a test. Saba stores this information in the database and provides view access to learners and their managers.

Communicating With Content

The following diagram explains the communication between an LMS and SCORM-compliant content through an API. For more details, see *Section 1.2* of SCORM RTE specification document.

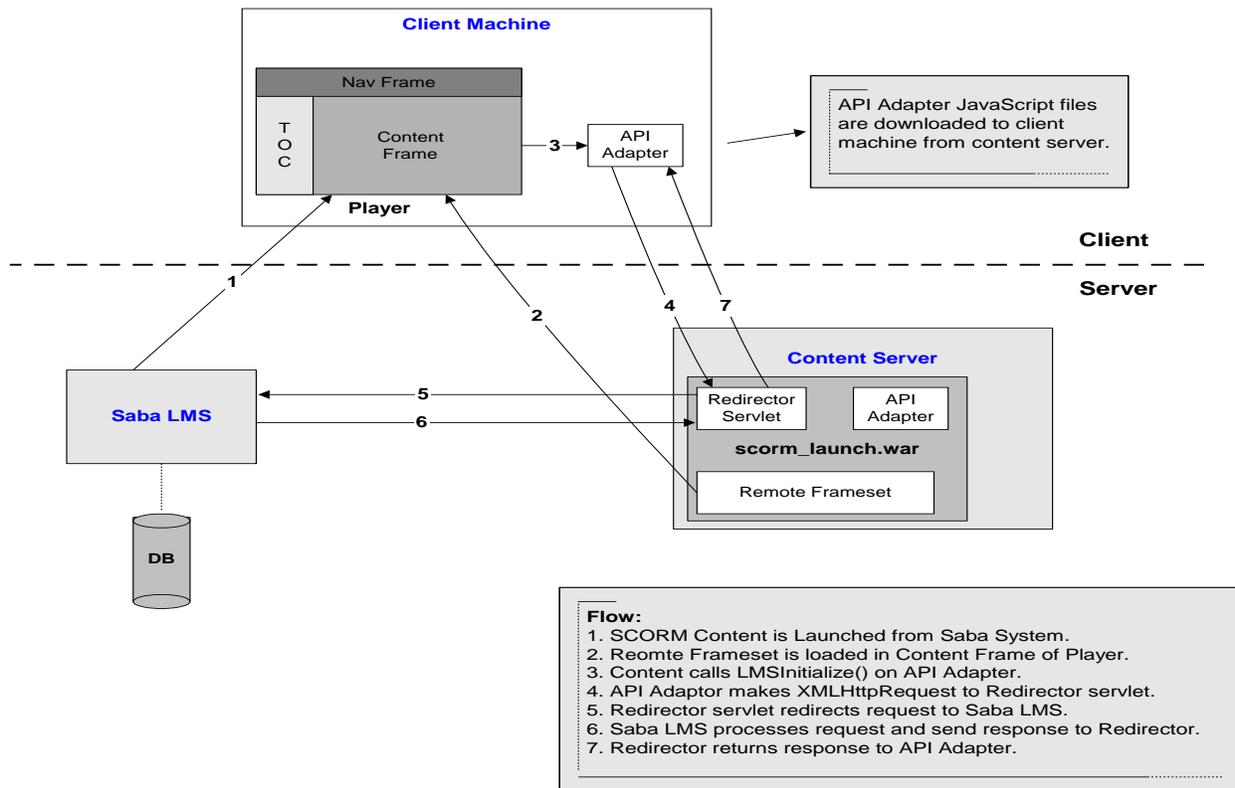
Figure 3: Communication between LMS and Content via API



Communicating With Content in Saba

The following diagram explains the communication between Saba LMS and SCORM-compliant content through the Content Server.

Figure 4: Communication between LMS and Content via Content Server



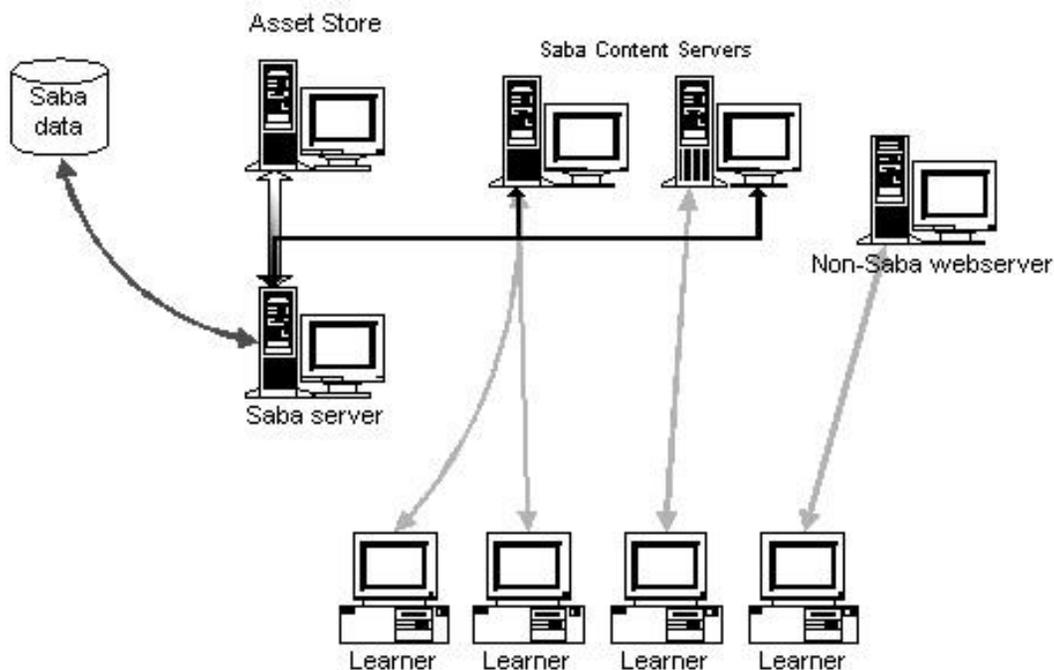
For more information, see [Content Working](#).

Content Storage Setup

Most Saba objects store their information in a relational database (the Saba data store). This database stores such things as employee information, class registrations, and Saba security privileges. However, content in Saba is stored differently, on a number of different computers:

The following figure illustrates how content is stored in Saba.

Figure 5: Saba Content Storage



- **Saba Content Repository**
The Saba content repository – a part of Saba LMS – stores information about the content, such as the content object’s name, the server it is stored on, and which classes have subscribed to the content. The repository also arranges content into a hierarchy of folders, so catalog administrators can find the content they need quickly. This information is stored in the Saba data store and can be accessed directly only by the Saba server; users and administrators connect with the Saba server, which in turn connects with the data store to fetch content repository information.
- **Saba Content Server**
The content is actually deployed on content servers. When users consume the content, their browsers make a direct connection with the content server. There are two kinds of content servers:
 - o A Web server can be registered with the Saba content administrator, making it a Saba content server. When you upload certain kinds of data, Saba deploys the appropriate files

to the Saba content server you choose. Saba is responsible for writing the appropriate files to that machine.

- o A Web server can also serve content without being registered with Saba (an external content server). In that case, when you import content into Saba, you will simply provide references to the data on the appropriate Web Server. You are responsible for putting the appropriate files on that Web Server.

Note that the same machine can serve as both a Saba content server and an external content server. When you register a Saba content server, you designate a certain directory on the server as being used for Saba content; Saba will be responsible for writing files to that directory. There is nothing to prevent you from using any other location on that Web Server for other content.

- Saba Asset Store

The asset store maintains exact copies of all the content files that are uploaded. Some content files have to be processed before they can be used by learners; for example, some content is uploaded as a ZIP archive, which has to be expanded before learners can use the content. Saba stores the exact file, unaltered, in its asset store; this lets Saba export the information when necessary, or re-deploy it to a new server. The Saba server is the only machine that needs to connect with the asset store; it archives content files there when they are uploaded, and fetches them off that machine when they are needed. The asset store acts as a safety vault for storing a copy of the content in a secure place from which it can be retrieved in the event the content servers on which it is deployed get corrupted, moved, or otherwise compromised.

Configuration

Before you can use SCORM-compliant content with Saba, you need to make certain configuration settings to the content as well as the different components of Saba Content.

This section describes the configuration for the following:

- [Global Settings](#)
- [Player Templates](#)
- [Content Level](#)

Global Settings

To work with content, you need to configure the Saba system by editing the property file for Content. The following table lists the various content properties.

Table 3: Content Properties

Classification	Property	Description	Notes
SCORM Standard	SCORM 1.2 Conformance	This property is used to manage conformance enforcement for SCORM 1.2 content. If it is set to conform (1), then the player checks the data model elements for correctness.	System wide
	Saba Application URL used by end users	This property is used when using a Reverse Proxy Server with Saba server. It contains protocol, hostname, and port number. For example, <a href="http://<servername>:<portnumber>">http://<servername>:<portnumber> When not in use, the property should be left blank.	System wide
	Chunk Size for file downloading	This property is used to specify the chunk size in Megabytes (MB). It is recommended to keep it less than 4 MB.	System wide
SCORM 2004 Support	SCORM 2004 Conformance	This property is used to manage conformance enforcement for SCORM 2004 content. If it is set to conform (1), then the player checks the data model elements for correctness.	System wide

	Table of Contents Visibility	This switch governs if the complete TOC should be visible even if sequencer determines some TOC elements to be invisible.	
Debugging Support	Enable SCORM API debug	This property is used for debugging SCORM Adapter (JavaScript) API. Options: <ul style="list-style-type: none"> • 1=Enable. It switches on the alert-based debug mode. • 0=Disable. This setting must be used in the production environment.	System wide
	Saba Application URL used by content server	This property is used when the content server needs a different URL than the one used by end users to communicate with the Saba server. If it is not used, then it should be left blank.	System wide
AICC Standard	AICC Logout Command Exit Behavior	This property is used to define the exit behavior for AICC logout. Options: <ul style="list-style-type: none"> • 0=Exit Content Only • 1=Exit both Content and Saba Application 	System wide
	Private Key File Location	This property is used to secure the content server. You need to create a key pair using “com.saba.secureproxy.KeyUtil” and place the path in this property.	System wide
	Bypass Content Results reporting call validation	This property is used for bypassing the Content results reporting call validation. Options: <ul style="list-style-type: none"> • 1=Bypass validation of Content Results reporting call. • 0=All Content Results reporting calls will be validated 	System wide
	Secure Content Server Timeout	This property is used for setting a valid session time (in minutes) during which the secure content session stays valid after the last HTTP request.	System wide
	Temporary Directory for file	This property is used to set the temporary directory used by all Saba content services/components. In a clustered	System wide

	upload and download	environment, you need to specify a shared network directory that is accessible by all the nodes in the cluster.	
--	---------------------	---	--

Player Templates

The Player Templates provide uniformity to the look and feel of the content delivered through the system. These templates control the visual presentation of the Saba Online Player at real time, and enforce a level of standardization for the content delivered using the Saba Online Player.

The Saba application allows you to create your own player templates with a choice of colors and fonts, header and footer location, and options to display or hide the navigation bar and table of contents.

When the administrator enables the Saba Content Management functionality on your system, the Saba application allows you to configure player templates and associate them with content.

The content administrators and content developers associate player templates with content during the following scenarios:

- Importing content into the Production Repository or Knowledge Base.
- Publishing content to the Production Repository or Knowledge Base using the Saba Content Manager

When the Saba Content Management module is disabled, the Saba application does not allow you to create new player templates. Therefore, the **New Player Template** link is not displayed.

However, it allows you to edit the properties of the following existing player templates:

- [System-Defined Templates](#)
- [Empty Player Templates](#)
- [3.x or 5.1 Compatible Player Templates](#)

While publishing content or importing content, you can associate any one of the player templates with content. In the **Player Template** field, use the picker to select the player template.

System-Defined Templates

These are standard templates provided by Saba. The system-defined template displays the table of contents and navigation controls. The player using this template is designed for multiple SCO/AU content.

Empty Player Templates

These templates do not display the table of contents or any navigation controls. The player using this template is designed for single SCO/AU content.

3.x or 5.1 Compatible Player Template

The player using this template allows content to pop up in its own window, and is designed for content that cannot run inside a frameset, uses third party cookies, or may need to resize the parent window to its own dimensions.

Player Template Properties

The following table describes the properties used by the different player templates.

Table 4: Player Template Properties

Property Type	Property Name	Description	Default Value		
			System-Defined Template	Empty Player Template	3x or 5.1 Compatibility Template
Look and Feel Properties	Height	Height of the overall player. This property comes into immediate effect once the player is launched. This is decided in “PlayerDispatcher.rdf” that resizes the player.	668	668	400
	Width	Width of the overall player. This property comes into immediate effect once the player is launched. This is decided in “PlayerDispatcher.rdf” that resizes the player.	924	924	500
	Theme	The theme governs the CSS for the player.	Default Online Player Theme	Default Online Player Theme	Default Online Player Theme
	TOC Width	The TOC area width. This property creates the TOC frame with this width and leaves the rest of the width for other frames. This is decided in the “FramesetGenerationCommand.java” while creating the TOC frame.	200	200	200
	Show Table of Contents	This property decides if the administrator wants to display the TOC area or not. This is decided in the “FramesetGenerationCommand.java” while creating the TOC frame.	True	False	True

	Show Navigation Controls	This property decides if the administrator wants to display the navigation area or not. This is decided in the “FramesetGenerationCommand.java” while creating the navigation frame.	True	False	False
	TOC Scrollable	This property governs the TOC scrollable behavior. This is decided in the “FramesetGenerationCommand.java” while creating the TOC frame.	False	False	False
	Display Header	This property governs the display of the customized header on top of the navigation area.	True	False	False
	Display Footer	This property governs the display of the customized footer on top of the navigation area.	True	False	False
Other Properties	Is Default	This property marks the default player template in the system.	True	False	False
Navigation Properties	Close Player on SCORM 1.2 Content Completion	This property is applicable only to SCORM 1.2 content. It governs the player close behavior. Once the SCORM 1.2 content is completed, it automatically closes the player. For more information, see Navigation Properties .	False	-	-
	Use Auto Navigation for SCORM 1.2 Content	This property is applicable only to SCORM 1.2 content. It governs the navigation behavior of SCORM 1.2 content. Once a SCO is completed, the content automatically navigates the learner to the next SCO. For more information, see Navigation Properties .	False	-	-

Navigation Properties

The following new navigation options have been added to the Player Templates:

- Close Player on SCORM 1.2 Content Completion
Upon receipt of the LMSFinish command, the Saba application automatically closes the Player window. This feature works well with legacy content that, on an earlier system, may have had control over the window it was playing in. If that content is running in the System Defined or Empty Players, then the content does not control the window. The content simply exits and displays the message “Please Choose a Navigation Option” to learners. With this option selected, the Saba application closes the player window on receipt of the LMSFinish command from the final SCO in the SCORM 1.2 content package.

- Use Auto Navigation for SCORM 1.2 Content
In a multi-SCO content, once the Saba application receives the LMSFinish command from a SCO, it automatically loads the next SCO. This option removes the requirement for learner to use the Player navigation and Table of Contents (TOC) to navigate between SCOs. Furthermore, it addresses the Instructional Design arguments that have taken place in the e-Learning content world over the confusion of having navigational controls inside a SCO (Next/Previous/Home, and so on), and the LMS provided controls in the form of a TOC and navigation bar.

Content Level

The Saba application provides the following options on the **Content Inventory Detail** screen:

- [Is Scoring](#)
- [Hide Exit Options \(Only for SCORM 2004 Content\)](#)
- [Hide Exit Without Saving](#)
- [Discard Attempts](#)

As a content administrator, you can select the these options only for the following content formats:

- AICC
- SCORM Package
- Deployed SCORM

Is Scoring

For a content inventory item, content administrators can enable the mastery score override by selecting the **Is Scoring** check box. This enables catalog administrators to edit and override the mastery score for content at the offering level. As a result, they can change the completion threshold for an offering.

Note: The mastery score is a score for the entire content package and is not applicable for individual SCOs within a package.

To enable the mastery score override:

1. Click **Content Administration > Repositories**.
2. Click the **Production Repository** link.
The **Production Repository** page appears.
3. Click the **Browse** tab.
4. Select and edit the required content.
The **Content Inventory Details** page appears.

Note: The content must belong to AICC, SCORM Package or Deployed SCORM format.

5. Select the **Is Scoring** check box.
6. Click the **Save** button.

Once you enable the mastery score override, catalog administrators can modify the value of the mastery score.

Note: The **Is Scoring** setting at content inventory affects the macro rollup of the offering. Therefore, for the content that never sends the score, **Is Scoring** bit should not be checked. Else, automatic rollup of offering can never happen. For information about the impact of **Is Scoring** setting, see [Macro Rollup](#).

If the content has a mastery score specified in the **imsmanifest.xml** file, then the **Is Scoring** check box is automatically selected.

For example, consider a case where a single SCO content item has a mastery score of 60 specified in the **imsmanifest.xml**. and this score is overridden at the offering level with a mastery score of 70. If a learner makes an attempt at this content and scores 65, then the lesson status still remains incomplete. In this example, the Saba application overrides the lesson status and marks it as passed at the SCO level. However, since it has not exceeded the mastery score of 70 at the offering level, the content module is marked as failed.

Hide Exit Options (Only for SCORM 2004 Content)

Saba online player provides a navigation bar with options to close the content gracefully. When content is closed using the navigation bar close button or the browser close button, the application provides **Exit Options** to the user. This may lead to confusion among users.

To resolve this issue, the **Exit Options** are presented to the user on browser close as well.

There are certain content that can handle abrupt closure scenarios by issuing the appropriate navigation request before closing. For such content, providing these **Exit Options** is not necessary. Therefore, the **Exit Options** are made configurable at the content level.

Configuration

On the **Content Inventory Detail** page, the application displays the **Hide Exit Options** check box. By default, this check box is unchecked.

The **Hide Exit Options** check box is applicable to SCORM 2004 content and visible only in the Edit mode.

Impact

On enabling **Hide Exit Options**, the content does not display the **Exit Options** popup for learners and administrators when they exit content; that is, when they close the player using the close option on the navigation bar or the browser.

The `<hideLMSUI>` elements in the content **imsmanifest.xml** are supported for *exitAll*, *suspendAll*, and *abandonAll* values only when the **Hide Exit Options** option is disabled.

Hide Exit Without Saving Option (Only for SCORM 2004 Content)

The **Hide Exit Without Saving** option enables administrators to disable the **Exit without saving** option for learners taking the content. It helps to prohibit users from discarding any content attempts that they make.

Configuration

On the **Content Inventory Detail** page, the application provides the **Hide Exit Without Saving** check box. By default, this check box is unchecked.

The **Hide Exit Without Saving** check box is applicable to SCORM 2004 content and visible only in the Edit mode.

Impact

The **Hide Exit Without Saving** option takes effect only when the **Hide Exit Options** is disabled. On enabling the **Hide Exit Without Saving** option, the content does not display the **Exit without saving** option for learners and administrators in the **Exit Options** popup when they exit content; that is, when they close the player using the close option on the navigation bar or the browser.

If a learner selects the **Exit without saving** option when taking content, all results and progress are lost and the learner must take the content all over again from the start.

For any combination of `<hideLMSUI>` and **Hide Exit Without Saving**, if all three options are disabled, the **Exit Options** popup is not displayed.

If none of the **Exit Options** are visible due to any of the above settings or `<hideLMSUI>` settings in the content, the player by default issues a *SuspendAll* navigation request on player closure when all of the following conditions are satisfied:

- Content navigation is not completed and subsequently, the player is closed.
- Current SCO is not yet unloaded from the player.

Content can send a navigation request before closing the player. In such a case, the default navigation request *SuspendAll* that is sent by the player, takes precedence.

Discard Current Attempts

The **Discard current attempt** button enables the content administrator to discard any currently active attempt in the Preview mode and start a fresh preview. If there is no active attempt, the link silently returns without any error.

This feature is available only in the Preview mode from the **Content Details** page. If the exit criteria popup box is suppressed by selecting the **Hide Exit Options** check box, then the content administrator has no means available to abandon progress and start a fresh preview for testing purposes. The **Discard current attempt** button allows the content administrators to re-launch content afresh.

The **Discard current attempt** button is available to the content administrator irrespective of any setting in *<hideLMSUI>* in the **imsmanifest.xml** of the content and irrespective of **Hide Exit Options** or **Hide Exit without Saving** options on the content inventory.

Content Management

This section describes the following content management activities:

- [Importing Content](#)
- [Updating Content](#)
- [Removing Content](#)

Importing Content

The content import process involves the following:

- Validation of the content manifest
- Creating entry in the Production Repository
- Generating TOC and saving it
- Copying the initial file/URL to the Asset Store
- If deployment is required, then deploying content to the Content Server
- Reading the set of transformation files on the Application Server, whose location is pointed to by:
 - o Content Properties. Server URL + SabaWeb Properties.Assets URL

On the **Import Content** page, you need to provide content format-specific information about the content that you are importing. You need to provide the location of a file or directory that you want to upload to the content store. You can also provide information for any additional data fields defined by your system administrator.

There are two broad categories of contents:

- Contents available as physical files (deployed on Saba Content Server}
Includes SCORM 1.2, SCORM 2004, File, Directory, and IMS content types.
- URL based contents (not deployed on Saba Content Server}
AICC URL, Deployed SCORM, and URL contents types.

The SCORM, Deployed SCORM, and IMS contents types contain a manifest file that describes the structure of content along with other Meta details. For AICC contents, the course structure files (.au, .crs, .cst, .des) contain this information. Whereas, for File, Directory, AICC URL, and URL contents, there is no structure detail.

Import process creates a normalized manifest file for all types of contents. This allows Player and Sequencing Engine to treat all types of contents in uniform manner.

After entering all the necessary information, click **Import** to upload the content to the Saba content store. Once you finish importing the content, Saba displays a summary page describing the content

object. Additionally, you can preview the content in the Saba content browser by clicking the **Preview Content** link from the **Import Content** page.

Once you have imported the content, learning offerings can subscribe to it, thereby making the content available to learners.

Updating Content

The process to update content consists of the following:

- **Updating Content**
Content Administrators can decide if the existing runtime information should be cleaned or not. The runtime information includes Learner Results, Sequencing Information of any incomplete attempts, and Error logs. If the updated content has some structural changes (in terms of SCO's), then the administrators can decide to clean the runtime data to avoid improper behavior, thereby completing the content. If there are no structural changes, then it is advisable to retain the existing runtime information.
- **Updating Content Metadata**
The Content Metadata changes do not lead to change in the content runtime behavior. However, the updated fields like 'isScoring' can lead to content completion behavior changes.

Removing Content

Saba allows you to remove only the content that is not subscribed to by any offering. On removal of the content, physical files are deleted from the Content Server. The runtime data is also cleaned from the SABA LMS.

Content Working

Content Launch

This section describes how the SCO-launched content obtains the handle to the SCORM API adapter.

As illustrated in [Communicating With Content in Saba](#), when the player is launched from the application, apart from loading the navigation controls in the TOC and Navigation Bar sections, the following frameset is also loaded in the Content Area.

```
<html>
  <head>
    <title>Saba Content Player</title>
    <script language="javascript">
      var API = null;
      var API_1481_11 = null;
    </script>
    ....
  </head>
  <frameset name='scormFrameset' rows='*,1' framespacing='0'
  frameborder='no'>
    <frame name='sco' src='loading.html' border='0'
    noresize/>
    <frame name='scorm_api'
    src='scorm_frame.html?query_string' border='0'
    noresize/>
  </frameset>
</html>
```

The **'query_string'** contains the actual SCO URL, learning context, and other attributes passed by the LMS to the SCORM adapter.

The **'scorm_frame.html'** is responsible for loading the JavaScript-based SCORM adapter and for setting the **API** and **API_1481_11** (for SCORM 2004) references.

Once the API adapter is set, the **'scorm_frame.html'** loads the SCO in the adjacent frame using the SCO URL. This ensures that the adapter is loaded first and then the SCO, thus ensuring that the adapter is always located by the SCO.

The SCORM Content can use the algorithm recommended in the SCORM specifications document to find the API handle. For information, see *Section 3.3.6.1* of SCORM 1.2 RTE Specification or *Section 3.3.1* of SCORM 2004 RTE Specification.

Content Tracking

This section covers the tracking behaviors that are left to LMS implementation by the SCORM RTE specification.

Determination of lesson status for SCORM 1.2

Lesson status set by the content is overwritten by Saba LMS as per the details provided in the SCORM 1.2 RTE Specification in `cmi.core.lesson_status` description of Section 3.4.4.

cmi.credit	Mastery Score	Score from content	Lesson Status from content	Saba Derived Lesson Status
Credit	Undefined	Undefined	incomplete/ passed/ failed/ completed	Lesson status sent by the content.
Credit	Undefined	Defined	incomplete/ passed/ failed/ completed	Lesson status sent by the content
Credit	Defined	Undefined	incomplete/ passed/ failed/ completed	Lesson status sent by the content
Credit	Defined	Defined	incomplete/ passed/ failed/ completed	If(score >= mastery score) passed Else failed.

Locking for the lesson status and score

For a given learner consumption of the content, Saba locks down the best result for the SCO, ever sent by the content / derived by Saba in a single user session or in multiple user sessions.

Statuses are ordered from best to worst as follows:

passed > failed > completed > incomplete.

For example,

A SCO sends a status failed and a score 45 in the first LMSCommit.

The SCO later sends a status passed and a score 75 in the second LMSCommit. Saba accepts this status as passed > failed (last stored status).

The SCO further sends a status failed and a score 64 in the third LMSCommit. The status of the SCO remains unchanged as failed < passed (last stored status).

This logic holds good across SCO launches.

Determination of completion status for SCORM 2004

Saba follows SCORM 2004 RTE specification for evaluating the completion status. Please refer to the Section 4.2.4.1 Completion Status Evaluation of the specification.

There are slight differences in determination of completion status for Edition 2 and Edition 3 content.

Some of the differences are -

- In Edition 3, comparison between *completion threshold* and *progress measure* is corrected to include *equals*.
- In Edition 3, if *completion threshold* is defined and no *progress measure* is provided by the content, the LMS can't make any assumptions and the completion status evaluates to *unknown*; in Edition 2, LMS assumes the completion status provided by the content.
- Impact on Sequencing Tracking Model: In Edition 3, if the *completion status* is set to *not attempted*, then the *attempt progress status* shall be set to *true* and the *attempt completion status* shall be set to *false*; in Edition 2 the behavior is not defined.

Determination of success status for SCORM 2004

Saba follows SCORM 2004 RTE specification for evaluating the success status. Please refer to the Section 4.2.22.1 Success Status Evaluation of the specification.

Support for interactions

Saba supports both *jouneeling* as well as *status* scheme for recording interactions as defined by the Section 4.2.9 Interactions of SCORM 2004 RTE specification. It is the content author who decides which scheme to use in the content.

LMS is required to support smallest permitted maximum 250 sets of interaction data. Saba doesn't put any restriction on the recording of interaction data.

What is not supported in Saba?

Following things are not supported in Saba:

- Multiple organizations: Saba simply supports the default organization defined in the *imsmanifest.xml*. Rest of the organizations if defined are ignored by Saba.
- Cmi.core.credit: Saba supports only *credit* value for this data model element; it doesn't support *no-credit* consumption of the content.
- Cmi.core.lesson_mode: Saba supports only *normal* lesson mode. It does not support *browse* or *review* lesson modes.

Recommendations for content vendors:

DOs:

- SCO should be designed in such a way that no critical data is lost if the player is abruptly or unexpectedly closed.
- SCO should spread LMSCommit() calls throughout the session while dealing with large volume of data. It should be judicious enough not to burden LMS with frequent calls to LMSCommit() if data is not changing that frequently.

DON'Ts:

- SCO should not try to close the window it has not created. If the SCO is designed to open in a new window, it should close the window on termination. Whether opened in a new window or inside a player window, it should never attempt to close the player window.
- If a SCO is designed to use cmi.credit and / or lesson_mode values to take different paths within the SCO, it will not work in Saba as it doesn't support any other values than *credit* and *normal* respectively.
- As there is locking on lesson status and score, implementing negative marking system within a SCO may not work if score is frequently set. In that case score should be set only when final score is arrived.

Content Sequencing

Following is the example of SCORM 2004 Ed 3 Sequencing model for said behavior -

Learning Behavior:

A learner is asked to take a pre test.

If he qualifies it, he is not required to go through the learning material. Result of the pre test is taken as the final result of the learner.

If he doesn't qualify the pre test, he is required to go through the learning material spread in one or more lessons.

Finally the learner needs to qualify the post test to prove his competencies in the just learnt material.

Result of post test is taken as the final result for the learner.

Recommended Sequencing Model:

Attached Imsmanifest.xml file exhibits the above mentioned learning behavior.



Content Rollup

The process of Content Rollup involves:

- Checking the Lesson Status of the Lessons in a Content
 - Derive if the Content has been completed as per the completion criteria for Passed / Failed / Completed.
- If the Content has met the completion criteria then the content completion status is marked and a check for completion of the Offering is done.
- If all the Content modules required for completion in a given path in the Offering have been completed then the Offering is marked complete.
- Offering completion is subject to the Auto Completion BR being enabled at the Offering.

The content rollup process is described in detail in the following sections:

- [Completion Logic](#)
- [Content Completion Logic](#)
- [Content Module Completion Logic](#)

Completion Logic

For each enrollment being reviewed for completion, the completion criteria for the offering as applicable to the registration is taken and each content module is evaluated for completion.

Content Completion Logic

The attempt data across all the lessons in the content are used in the process to check for completion, the attempt data is ordered by the attempt date; this provides for computation of the correct completion date of the content.

Lesson Status Computation Case

Note: The logic explained in this section is irrespective of whether the Content is set as Scoring or Not Scoring.

Content reports a lesson status of “Completed” and passes a SCORE; if a Mastery Score is present. Then it is further used to compute Lesson Status as Passed or Failed.

Note: Mastery Score is available only if the content manifest has a mastery score declared in it and the manifest is used in the import process

The following table lists the various statuses for content completion with respect to the statuses of all lessons in the content.

Table 5: Statuses of Content with respect to Status of Lessons

Lesson Status of all Lessons in the Content	Content Completion Status
Passed	Passed
Failed	Failed
Completed	Completed
Passed / Completed	Passed
Failed / Completed	Failed
Any other combination	In Progress

Content Module Completion Logic

The following table lists the combination of various parameters that derive the content module completion status.

Table 6: Content Module Completion Status Logic

isScoring	Mastery Score	Learner Score	Content Status	Completion Status
False	Not Applicable	Not Considered	Completed	Completed Successfully
False	Not Applicable	Not Considered	Passed	Completed Successfully
False	Not Applicable	Not Considered	Failed	Completed Successfully
True	Not Specified	Present	Completed	Not Evaluated
True	Not Specified	Present	Passed	
True	Not Specified	Present	Failed	
True	Not Specified	Not Present	Completed	Not Evaluated
True	Not Specified	Not Present	Passed	
True	Not Specified	Not Present	Failed	
True	Specified	Present	Completed	Completed Successfully if Learner score \geq Mastery Score Completed Unsuccessfully if Learner Score $<$ Mastery Score

True	Specified	Present	Passed	Completed Successfully if Learner score \geq Mastery Score Completed Unsuccessfully if Learner Score $<$ Mastery Score
True	Specified	Present	Failed	Completed Successfully if Learner score \geq Mastery Score Completed Unsuccessfully if Learner Score $<$ Mastery Score

Content Rollup Failure

Failure Symptoms

- General Exception in the Player
- Score or Lesson Status not captured in Saba
- Offering not completed

Failure Diagnosis

- Communication failure between Player – SCORM Bridge – Saba LMS
- Failure in processing of content request at Saba end
- Missing *exitAU* command results in completion check not being performed.
- Auto Completion BR has been Switched OFF at Offering Level
- Failure in cluster environment

SCO Completion (For SCORM 2004 Contents)

Saba follows the SCORM RTE specification for SCO completion.

For information, see *Section 4.2.4.1* of the SCORM RTE specification document.

Micro Rollup (Content Module Level)

After the Saba application provided support for SCORM 2004 3rd Edition, there have been changes in the content rollup logic used by Saba.

The following sections describe these changes in content rollup logic:

- [Before SCORM 2004 3rd Edition Support](#)
- [After SCORM 2004 3rd Edition Support](#)

Before SCORM 2004 3rd Edition Support

This section includes the following:

- [Trigger for Content Rollup](#)
- [Content Rollup Logic](#)

Trigger for Content Rollup

Unlike AICC and SCORM 1.2 content, where exit (ExitAU/LMSFinish) of every content item (AU/SCO) triggers content rollup, the Sequencer triggers the content rollup for SCORM 2004 contents.

The content rollup is triggered when:

- Sequencer returns Course Complete or Exit Session state
- When *Exit All* or *Abandon All* navigation request is triggered.

Content Rollup Logic

The content rollup logic consists of:

- [Status Rollup](#)
- [Score Rollup](#)

Status Rollup

SCORM 2004 content has two separate data model elements for mastery and participation. These elements are “cmi.success_status” and “cmi.completion_status”.

During content rollup, the application derives single status for each SCO for rollup using completion status and success status as follows:

Table 7: SCO Status Rollup Logic

Completion Status	Success Status	SCO Status for Rollup
Unknown	Unknown	Not Evaluated
Unknown	Passed	Passed
Unknown	Failed	Failed
Completed	Unknown	Completed
Completed	Passed	Passed
Completed	Failed	Failed
Incomplete	Unknown	Incomplete
Incomplete	Passed	Passed

Incomplete	Failed	Failed
Not Attempted	Unknown	Not Attempted
Not Attempted	Passed	Passed
Not Attempted	Failed	Failed

For a single / multi-SCO content, the status of the content is calculated as follows:

- If all SCOs are *passed*, then the content is passed.
- If all SCOs are *completed*, then the content is completed.
- If all SCOs are *failed*, then the content is failed.
- If any or all SCOs are *not attempted*, then the content is *not attempted*.
- If all SCOs are either *passed* or *completed*, then the content is *passed*.
- If all SCOs are either *failed* or *completed*, then the content is *completed*.
- Else, the content is *incomplete*.

Note: The SCOs that have null values for completion status and success status are not attempted at all; they are skipped from the rollup calculation.

Score Rollup

The following score rollup is used:

$$\text{Content score} = \text{Sum of \% score of all SCOs} / \text{number of SCOs}$$

Note: SCOs that have null values for completion status and success status and the SCOs that have score as null are not used in this calculation.

After SCORM 2004 3rd Edition Support

For SCORM 2004 content, the sequencing engine according to the SCORM 2004 Sequencing and Navigation (SN) specification triggers the rollup. For more information, see *Section 3.7* and *Section 4.6* of the specification document.

Rollup is one of the sub-processes in the overall sequencing process invoked when the content or the player sends a navigation request. Saba plays no role in this calculation.

This section includes the following:

- [Content \(Attempt\) Completion](#)
- [Content Rollup Logic](#)

Content (Attempt) Completion

An attempt on the SCORM 2004 content is treated as complete in the following cases:

- Sequencer receives *Exit All* navigation request (user closes the content using **Exit and Finish** option).
- Sequencer returns *Course Complete* or *Exit Session* status on processing any navigation request.

Content Rollup Logic

Content statuses and score are directly taken from the rolled up statuses while the score of the root activity is taken from the sequencer.

Note: The content results are discarded when the *Abandon All* navigation request is triggered by the content or the player (when user closes the content using **Exit without saving** option).

Macro Rollup (Offering Level)

In the Saba application, you can have multiple contents (modules) associated with the offering. A maximum of two paths for completion are defined for an offering. A content can be required for the completion of any or both the paths.

This section includes the following:

- [Concepts](#)
- [Offering Level Rollup](#)
- [Module Status Derivation](#)

Concepts

The following table describes the common terms used in a macro rollup of content in Saba.

Table 8: Macro Rollup Terminology

Term	Description
Scoring Content	A content that has mastery score or progress measure defined for any SCO.
Module Status	The Saba derived status for a content module associated with the offering.
Offering Status	The status that results from the rollup of content module statuses.

Offering Level Rollup

The offering level rollup logic consists of:

- [Rollup Trigger](#)
- [Rollup Logic](#)

Rollup Trigger

A rollup is triggered every time an associated content is completed (exit type).

- For SCORM 2004 content, the content completion is determined by the way a user has exited the content.
- For SCORM 1.2 and AICC content, completion of every SCO triggers a content-level rollup to determine if the content is complete.
- For an external player (for example, Rustici), the player determines the completion of the content.

Rollup Logic

When a path is completed, the offering moves to transcript. For any path of completion, the following rollup logic is applied to calculate the offering status:

Table 9: Offering Status Calculation

Module Status	Offering Status
Not Evaluated (any module)	Not Complete
Successful (all modules)	Successful
Unsuccessful (all modules)	Unsuccessful

The Offering Score is calculated as follows:

Offering score = Total of scores of all modules that reported a score / Count of modules that reported score

Module Status Derivation

The following logic is applied for deriving the module status:

```

On content completion (exit type - completed),

If (scoring content)
{
  if (saba mastery score defined)
  {
    if (content score not known)
    {
      Module Status = Not Evaluated // Doesn't move to transcript
    }
    else if (content score >= saba mastery score)
    {

```

```
        Module Status = Successful
    }
    else if (content score < saba mastery score)
    {
        Module Status = Unsuccessful
    }
}
else
{
    Module Status = Success status of content
}
}
else
{
    if (success status of content == passed or failed)
        // Content is treated completed when success status is known.
        Module Status = Successful
    else if (completion status of content == completed)
        Module Status = Successful
    else
        Module Status = Not Evaluated
}
```

Problem Solving

This section describes the following problem-solving mechanisms:

- [Tools](#)
- [Saba Content Communication Log](#)
- [Common Problems](#)

Tools

The following tools can be used for solving issues around SCORM-compliant content:

- [Content Compliance Test Tool](#)
- [HTTP Capture Tools](#)

Content Compliance Test Tool

The Content Compliance Test (CCT) Tool is available to the content administrator under the **Tools** tab. This tool is used to validate import time and runtime part of SCORM 1.2 standard.

The content compliance test tool validates SCORM 1.2 content for:

- Manifest correctness
- Runtime correctness

The tool is also used to generate reports. These reports contain:

- Warnings
Addressing the warnings make the package is compliant to SCORM 1.2 standard. The content import or launch process is allowed even if there are warnings shown by compliance tool.
- Errors
The errors in the report indicate that the package is not compliant to SCORM 1.2 standard.

The following illustrates a CCT log:

```
(43073) Failed to import Content Inventory: Test243753837  
(43269) Invalid content. (*)ERROR: Failed to extract the  
manifest file from content package. Possible cause: The  
markup in XML document (manifest file) preceding the root  
element is not well-formed.
```

HTTP Capture Tools

The HTTP Capture tools include:

- [HTTP Sniffer](#)
- [Ethereal](#)
- [TCP Tunnel](#)

HTTP Sniffer

With HTTP Sniffer, you can view all browser HTTP requests and server HTTP responses. The HTTP Sniffer allows web developers and network administrators to intercept, view, and analyze all the HTTP traffic between a web browser or any program that uses the HTTP protocol and the web server.

Ethereal

Ethereal is used by network professionals around the world for troubleshooting, analysis, software and protocol development, and education. It has all of the standard features you would expect in a protocol analyzer.

For more information, see <http://www.ethereal.com>.

TCP Tunnel

TCP Tunnel is a command line tool that allows you to forward TCP connections made to your computer. TCP Tunnel redirects TCP connections from a local port to a remote IP and port. Usually, TCP Tunnel forwards any TCP protocol. For example, HTTP, SMTP, POP, TELNET, NETBIOS, and SSL.

TCP Tunnel has HTTP proxy support. That is, connections can be tunneled through a variable number of HTTP proxies. This allows your software to use Internet services based on TCP/IP when there is a firewall that blocks outgoing connections. Also, if the proxies do not reveal your IP address, you will be able to make anonymous TCP connections!

Saba Content Communication Log

Once you preview the content, the communication log becomes available. The link to open this log is available below the **Preview Content** link.

This log captures all the runtime activity between content and LMS from launch till completion of the content. It also captures all data and calls passed between them.

This log should be used to determine the data that is passed by content and saved by the system. This can be verified using the result view of content administrators.

Common Problems

This section describes the various problems that occur while using SCORM-compliant content in Saba. Specifically, it addresses problems related to:

- [Content Import](#)
- [Content Communication](#)
- [Server Configuration](#)
- [Network Infrastructure](#)
- [Other](#)

Content Import

The content import process may not succeed due to the following reasons:

- Non-conformance of SCORM package
- Presence of multi-byte characters in **imsmanifest.xml** file

Generally, large content (having size greater than 20 MB) takes more time for import. This is also true for content with larger size of **imsmanifest.xml** file.

Content Communication

This section describes the various symptoms and diagnosis of failure in content communication.

Failure Symptoms

The following failure symptoms are observed:

- Content does not send *exitAU* when the user clicks the browser close icon to close the window
- Content sends commands but does not listen for the response from the LMS
- In SCORM
 - Calls to SCORM Bridge fail
 - Calls to Saba LMS from SCORM Bridge fail
- In AICC
 - Content unable to connect to Saba LMS
- In SSL
 - Missing or invalid SSL Certificate
- General

- o Invalid Saba Certificate on Saba LMS Server

Failure Diagnosis

The following diagnostic steps can be taken to detect failures in content communication:

For SCORM

- Inspection of Error logs on the content server, Web server logs, Saba Application logs on Saba LMS.
- Tracing calls from Player – SCORM Bridge – Saba LMS
- Enabling SCORM API debug, allows stepping through the flow.
To enable SCORM API debug:
 - a. Login as a system administrator.
 - b. Go to **Sites > Content**.
 - c. Set *Enable SCORM API debug* property to **1**.

For AICC

Inspection of debug logs generated by content (if available), network trace of calls from Content to Saba LMS.

Server Configuration

The common problems faced during server configuration are:

- Incorrect setup of Content Server
Content administrator can use tools to test content server configuration.
- Problems occur if the Content Server is moved from one system to another and if there are references of content metadata stored in Saba LMS database. This results into launch failures for deployed contents.
This problem can be resolved by re-importing the deployed contents so that references in the database are updated.

Network Infrastructure

The common problems faced in network infrastructure are:

- Cross domain communication
This issue arises when the LMS and the Content Server are in different domain. This problem can be resolved by placing both, the Content Server and LMS, in same domain.
- Heavy network load may also cause communication leakage between content and Saba LMS.

Frequently Asked Questions (FAQs)

Q 1) What is the largest packaged SCORM zip file you can import into Saba thru the UI?

Ans - Currently, the import process will accommodate average sizes up to 75 MB depending on network variables. Saba provides an alternative for importing larger size courses called Deployed SCORM, which removes the network limitations.

Q 2) How many organizations does Saba 5.3 SP2 support?

Ans - Saba supports **one** organization at this time. You can specify a default organization or Saba will treat the first organization as default if none are specified.

Q 3) What is the suspend behavior of Saba 5.3 SP2 when a SCORM course is closed?

Ans - If the SCO sets **CMI.EXIT** to *suspend*, Saba sets **CMI.ENTRY** to *resume*. By setting **CMI.EXIT** to *suspend*, the SCO indicates that the learner has exited with the intent of returning to the SCO at a later time. On the next launch of the SCO, Saba considers it as the resumption of the previously suspended attempt and not a new attempt. Hence, Saba returns the runtime data from the previously suspended learner attempt.

Q 4) How does Saba handle mastery scores?

Ans - The **Mastery score** is defined in the content metadata and passed at import time. Saba catalog administrators can override the mastery score at the content module level, or can set a mastery score if there is none. When a user completes a content object, Saba attempts to derive a status for the entire object:

- o If every lesson that reported a status, reports **complete** or **passed**, Saba assigns the object a status of **passed**.
- o If every lesson that reported a status, reports **failed**, Saba assigns the object a status of **failed**.

In any other situation, Saba assigns a status of **Not Evaluated**. For example, if a user passes one lesson and fails another, or if any of the lessons report a status of “incomplete”, “browsed”, or “not attempted”, Saba will give the class a status of **Not Evaluated**. In addition, Saba will average the scores for each lesson to produce a numerical score for the class.

Saba only averages the scores that are reported. While calculating the score, Saba discards any lessons with a **null** value for the score. Reporting a score of zero, instead of null, would wrongly bring down the average.

Q 5) Does Saba perform conformance testing on SCORM content at runtime?

Ans - Saba supports two modes for running SCORM content: strict conformance checking and loose conformance checking. If strict conformance checking is enabled, SCORM content must be strictly conformant in order to run when launched by learners. If conformance checking is disabled, Saba allows any SCORM content to be run. By default, conformance checking is disabled.

Q 6) How does Saba’s compliance testing tool differ from the ADL testing tool?

Ans - Saba's Content Compliance testing tool provides the following capabilities which are beyond the scope of the ADL Conformance Test Suite:

- o When content import fails, it is mostly due to non-conformant content packages. The ADL Conformance Test Suite for validating content packages does not always provide the specific details. Saba's Content Compliance Testing Tool enumerates all issues with the content package and manifest file.
- o In specific cases, Saba allows the import of content, even if the content does not strictly conform to the SCORM specification. For example, Saba allows the import of content in the following cases:
 - o The default organization is not specified in manifest file
 - o No title specified for organization element
 - o No title specified for item element

These content packages can be successfully imported into Saba, although they would fail in ADL Conformance Test Suite.

When the content sets incorrect runtime data, the Content Compliance Testing Tool provides a list of all possible values for error cases. The ADL Conformance Test Suite does not provide a list of valid values.

Q 7) How does Saba display the scores when multiple attempts are made on a single piece of content?

Ans - Saba will display the first successful score, regardless of any subsequent score or lesson status. Also, Saba will display the highest unsuccessful score.

For example:

I fail a content item with a score of 40. On the transcripts page I will see completion status = unsuccessful, score = 40.

I have another attempt and score 35 this time. On the transcripts page I will see completion status = unsuccessful, score = 40.

On my 3rd I fail it again with a score of 50. On the transcripts page I will see completion status = unsuccessful, score = 50.

On my 4th attempt I pass it with a score of 80. On the transcripts page I will see completion status = successful, score = 80.

On my 5th attempt I pass it again. On the transcripts page I will see completion status = successful, score = 80.

References

This document sources certain information from the following SCORM Specification documents:

- SCORM 2004 3rd Edition Overview Version 1.0, Advanced Distributed Learning, October 20, 2006 Available at: <http://www.adlnet.gov>.
- SCORM 2004 3rd Edition Content Aggregation Model Version 1.0, Advanced Distributed Learning, October 20, 2006 Available at: <http://www.adlnet.gov>.
- SCORM 2004 3rd Edition Run-Time Environment Version 1.0, Advanced Distributed Learning, October 20, 2006 Available at: <http://www.adlnet.gov>.
- SCORM 2004 3rd Edition Sequencing and Navigation Version 1.0, Advanced Distributed Learning, October 20, 2006 Available at: <http://www.adlnet.gov>.
- Saba *Content Administrator Guide*.
- Saba *Content Management Guide*.