SCORM

Simple Sequencing Templates & Models

1st Edition

2003-02-27

Credits and Redistribution Guidelines

SCORM Simple Sequencing Templates and Models is a publication of the Learning Systems Architecture Lab at Carnegie Mellon. This work was sponsored by the Technical Support Working Group of the Combating Terrorism Technology Support Office and was created as a service to the e-learning community in an attempt to further the adoption of the Sharable Content Object Reference Model (SCORM).

The Learning Systems Architecture Lab at Carnegie Mellon (licensor) permits others (licensees) to copy, distribute, display, and create hyperlinks to the *SCORM Best Practices Guide for Content Developers* (the work). In return, credit must be given to the licensor. Licensees may not use the work for commercial purposes, without the permission of the licensor. The licensor permits others to distribute derivative works under a license identical to the one that governs the licensor's work.

Except where otherwise noted, this work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/1.0 or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

Attribution   Noncommercial   Share Alike

For hyperlinks to this document, the appropriate URL is: http://www.lsal.cmu.edu/lsal/expertise/projects/developersguide/

If you are redistributing or hyper-linking to this document, please notify us so that we may keep you informed of the latest revisions to these documents. For more information, please contact us.

Learning Systems Architecture Lab   Voice: +1.412.268.5322
Carnegie Mellon University   Fax: +1.412.268.4772
700 Technology Drive   Email: lsal@cmu.edu
Pittsburgh, Pennsylvania, USA 15219   Web: http://www.lsal.cmu.edu/

Overview of Sequencing Templates and Models

The sequencing templates describe potential behaviors of SCOs according to various instructional design strategies. The templates are designed to assist you in structuring your content to comply with SCORM sequencing guidelines. Any template or combination of templates can be "overlaid" on or combined with another template, creating a more complex instructional strategy for a course or a lesson. Combining the templates provided here will provide you with viable sequencing models that you can adapt to meet your particular training and educational requirements. This document shows several models for more complex instructional strategies. Depending upon how you apply behaviors to the structures, you can achieve a variety of outcomes. These templates are not intended to be exhaustive, but they should help you begin to identify new ways in which you can construct SCORM content while adhering to sequencing guidelines, and the true intent of SCORM: creating *reusable, interoperable, durable,* and *accessible* instructional materials.

Each template or model includes a content structure diagram representing the template and the instructional strategy and sequencing rules for the template. The rules are presented in both non-technical language (called Behavior to describe what you want the student to experience) and technical language (called SCORM Function to describe what will be coded to enable the behavior). Designers can follow the behaviors in the templates provided, and developers and programmers can follow the SCORM Functions to program the sequencing commands specified by the designer. In some instances, the SCORM Function says "No Unique SCORM Function" for the programmers. This occurs because the designer specifies a behavior that is either internal to the SCO or is not impacted by SCORM. Several templates include multiple applications of the rules so you will understand that identical content structure diagrams (or courses, lessons, etc.) can be sequenced in numerous ways.

In this document and in the sequencing rules, we refer to halting the learning in training and requiring manual intervention by the instructor. You might want to use this type of an instructional strategy if you need to prevent the learner from seeing additional content because (1) they require face-to-face interaction with an instructor to ensure they have grasped the material, (2) they need assistance beyond that which is available in the remaining content, or (3) they will be unable to understand the remaining content without a strong understanding of the content they have completed. You can accomplish this by creating rules that result in the learner being prevented from seeing any SCO. This manual intervention would vary by LMS; it is not specified by SCORM.
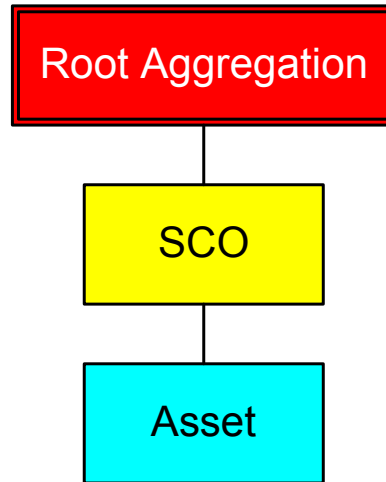
Template or Model	Description	Rule Applications
Template 1	Single SCO with a Single Asset	1
Template 2	Single SCO with Multiple Assets	1
Template 3	The Black Box; single SCO with multiple assets and complex internal structure	1
Template 4	Multiple SCOs with Assets	2
Template 5	Remediating Using Objectives	2
Template 6	Pre- and Post-Test Sequencing	1
Template 7	Pre- and Post-Test Sequencing (2)	1
Template 8	Remediating Using Objectives (2)	1
Template 9	Basic Three-way Branching	2
Template 10	Pre- and Post-Test Sequencing with New Content for Remediation	1
Model 1	Remediating Multiple Aggregations	2
Model 2	Mastery Testing Multiple Aggregations	1
Model 3	Pre- and Post-Test Sequencing with Aggregations	1
Model 4	Traditional CBT Branching with Multiple Decisions	1

Summary of Templates and Models

Any template or combination of templates can be "overlaid" on or combined with other templates, creating increasingly complex instructional strategies for courses or lessons. The models that follow show unique combinations of the templates presented in the previous section. The models show the reusability of the templates by labeling each as an instance of a template. In addition, the rules for each model specify from which template, as well as which application of the template, they were obtained. Depending upon how you apply behaviors and rules to the structures, you can achieve a variety of outcomes. These templates and models will provide you with viable sequencing options you can adapt to meet your particular training and educational requirements. For very complex instructional strategies, you can also apply any model or combination of models to another model as was done with the templates.

TEMPLATE 1:
Single SCO

```
Root Aggregation
       |
      SCO
       |
     Asset
```

© Copyright 2003, Carnegie Mellon University, Some Rights Reserved

Template 1 Rules:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must complete the SCO.	Rollup: All satisfied, completed

**TEMPLATE 2:
SCO with Assets**
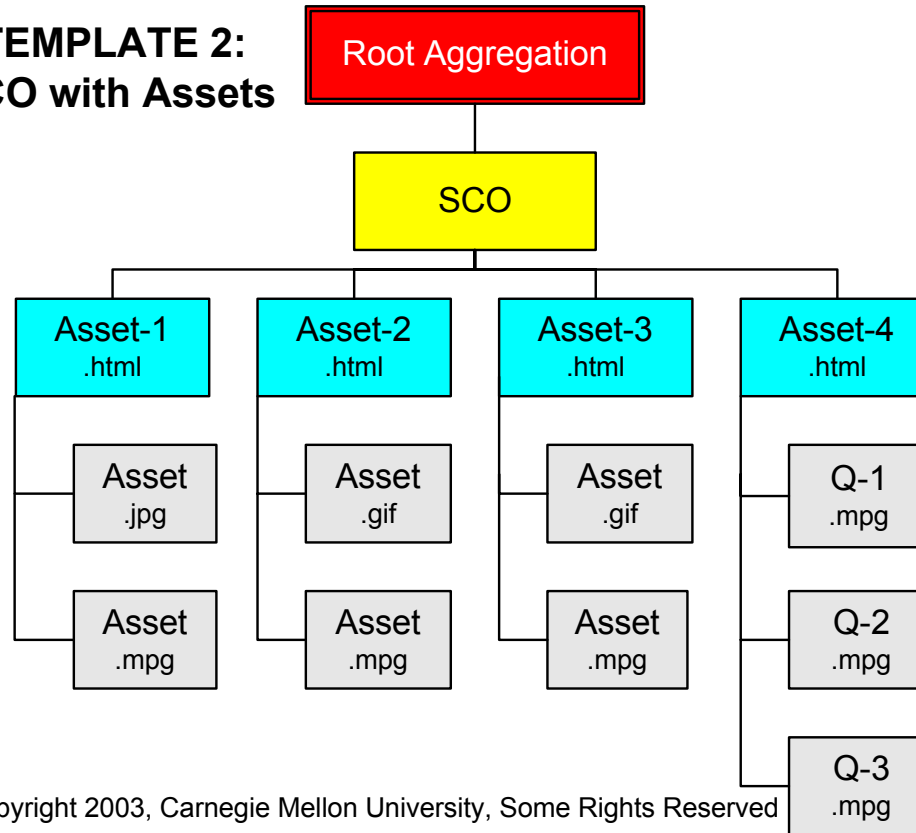
Template 2 Rules:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must complete the SCO.	Rollup: All: satisfied, completed
2. To complete the SCO, the learner must complete the assessment in Asset-4 within the SCO.	*No SCORM function*

TEMPLATE 3:
The Black Box

Root Aggregation

SCO

Asset-1

Asset-2

Asset-3

Asset-4
assessment

Asset
.jpg

Asset
.gif

Asset
.gif

Q-1
.mpg

Asset
.mpg

Asset
.mpg

Asset
.mpg

Q-2
.mpg

Q-3
.mpg
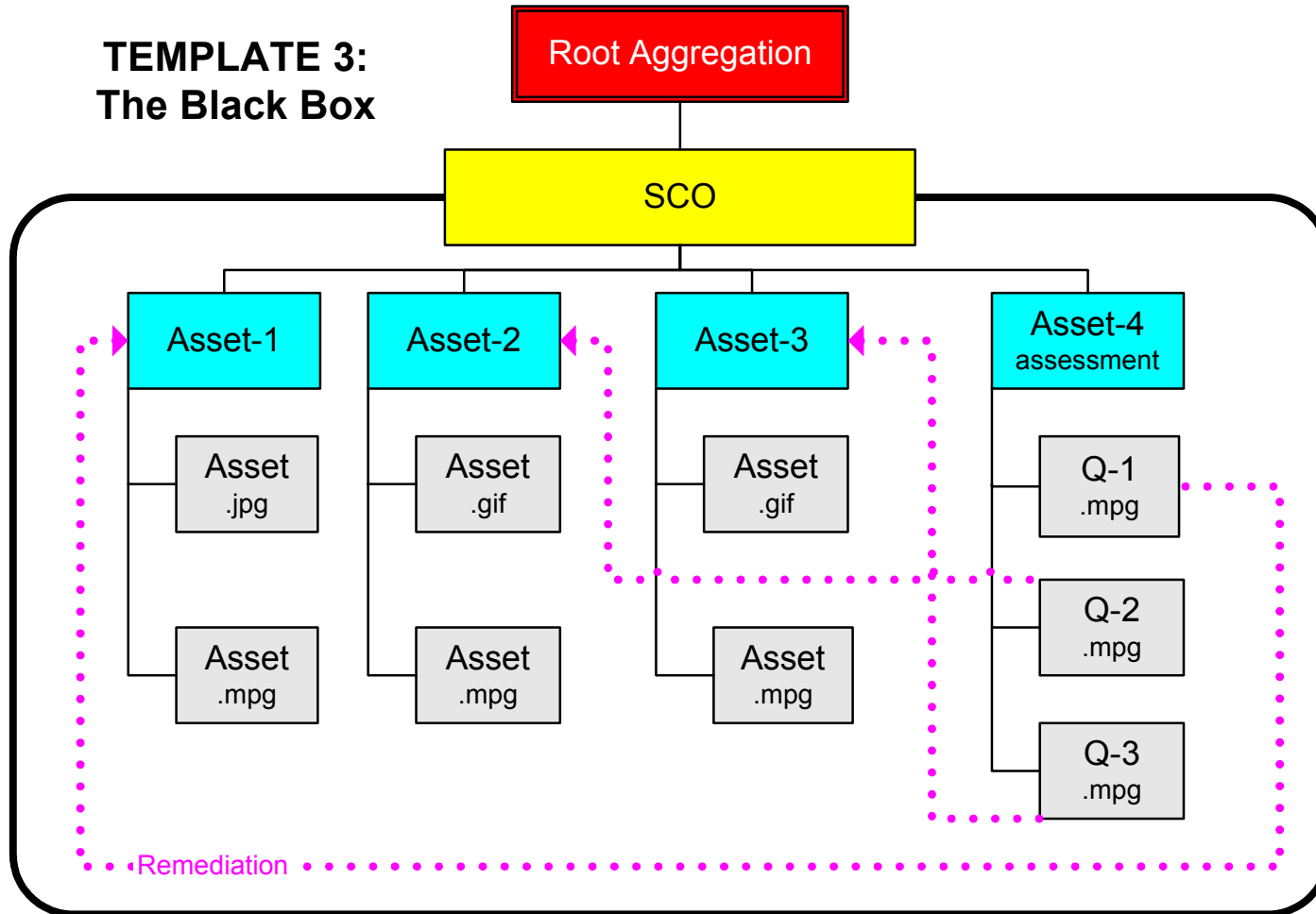
Remediation

All remediation occurs as intra-SCO branching, so there is no impact to inter-SCO sequencing.

© Copyright 2003, Carnegie Mellon University, Some Rights Reserved

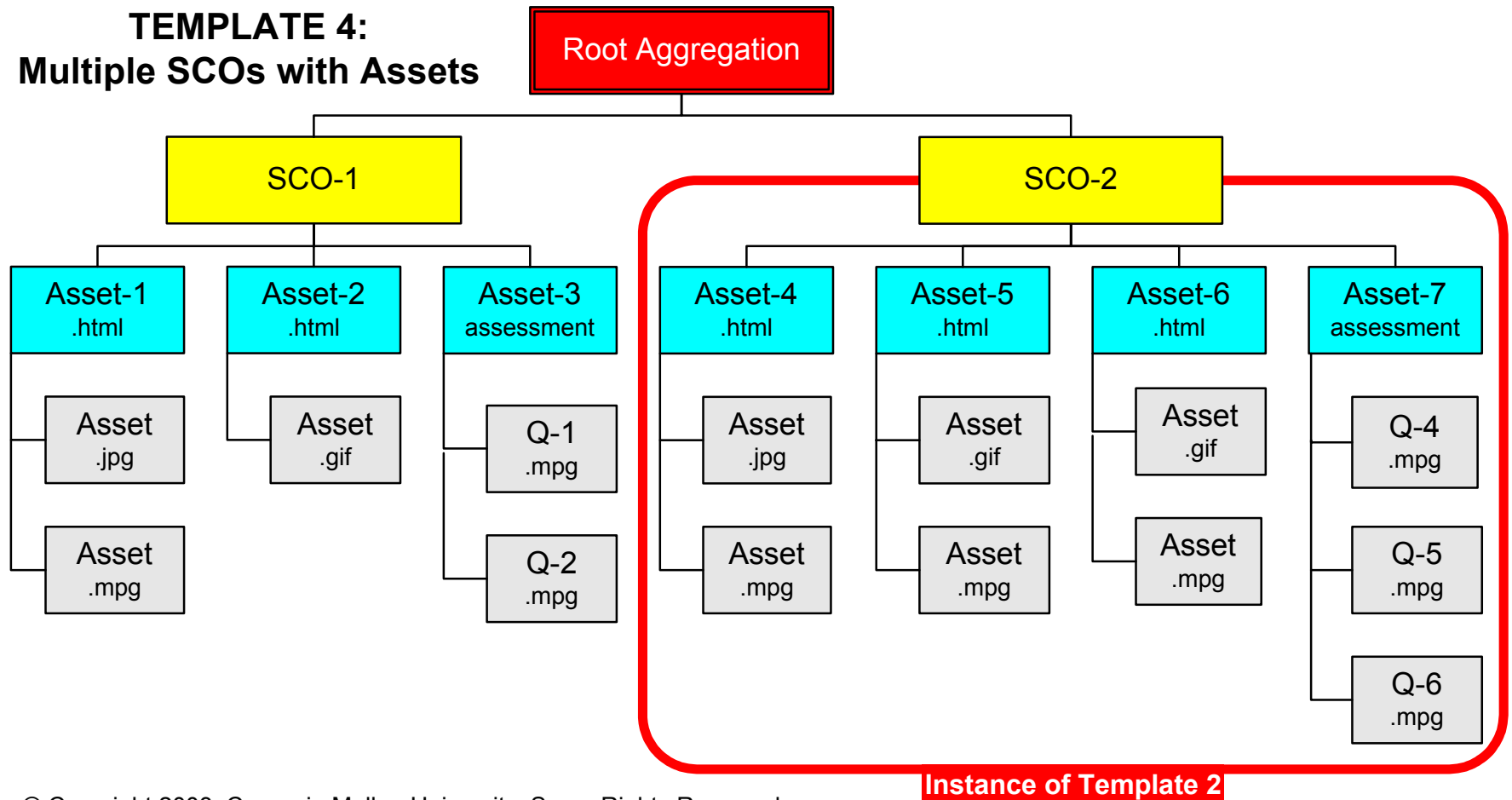| Template 3 Rules:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must complete the SCO.	Rollup: All: satisfied, completed

urn:1stEd.SCORMSS.LSAL.CMU.EDU: 512eb872-43ac-44db-b1e2-f604ec138239

**TEMPLATE 4:
Multiple SCOs with Assets**

Root Aggregation

SCO-1

SCO-2

Asset-1
.html

Asset-2
.html

Asset-3
assessment

Asset-4
.html

Asset-5
.html

Asset-6
.html

Asset-7
assessment

Asset
.jpg

Asset
.gif

Q-1
.mpg

Asset
.jpg

Asset
.gif

Asset
.gif

Q-4
.mpg

Asset
.mpg

Q-2
.mpg

Asset
.mpg

Asset
.mpg

Asset
.mpg

Q-5
.mpg

Q-6
.mpg

Instance of Template 2

Template 4 Rules *(Application A)*:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must complete SCO-1 and SCO-2.	Rollup: All: satisfied, completed.
2. To complete each SCO, the learner must complete the assessments within the SCOs.	*No SCORM function*
3. The learner cannot start SCO-2 until SCO-1 is complete.	SCO-1: If Not complete, Deny Forward Progress
4. The learner can return to SCO-1 from SCO-2 at any time.	Root Aggregation: Forward Only=false

Template 4 Rules *(Application B)*:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must complete SCO-1 and SCO-2.	Rollup: All: satisfied, completed.
2. To complete each SCO, the learner must complete the assessments within the SCOs.	*No SCORM function*
3. The learner can view the SCOs in any order.	Root Aggregation: Flow and Choice

TEMPLATE 5: Remediating Using Objectives

Set OBJ   Read OBJ

© Copyright 2003, Carnegie Mellon University, Some Rights Reserved
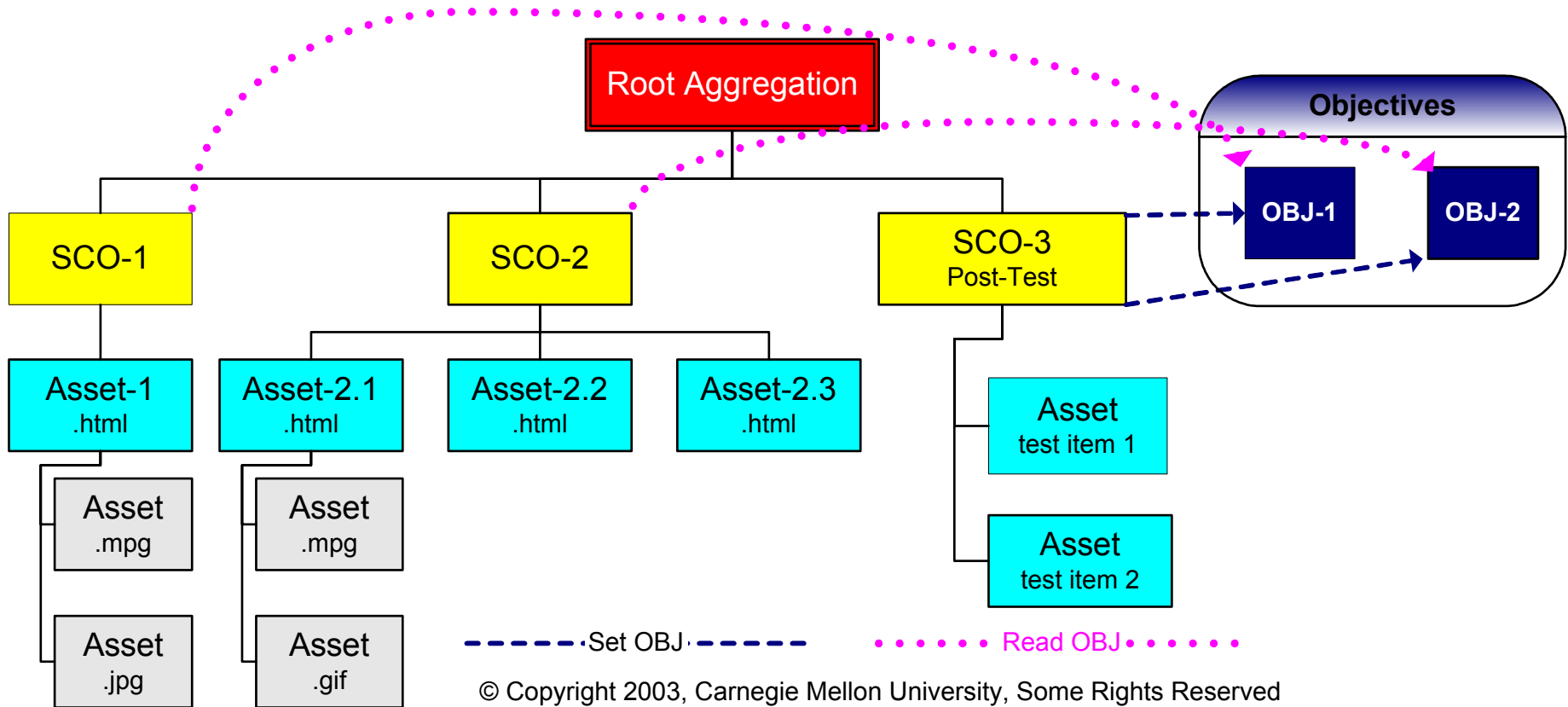
SCORM Simple Sequencing Templates & Models

Template 5 Rules *(Application A)*:

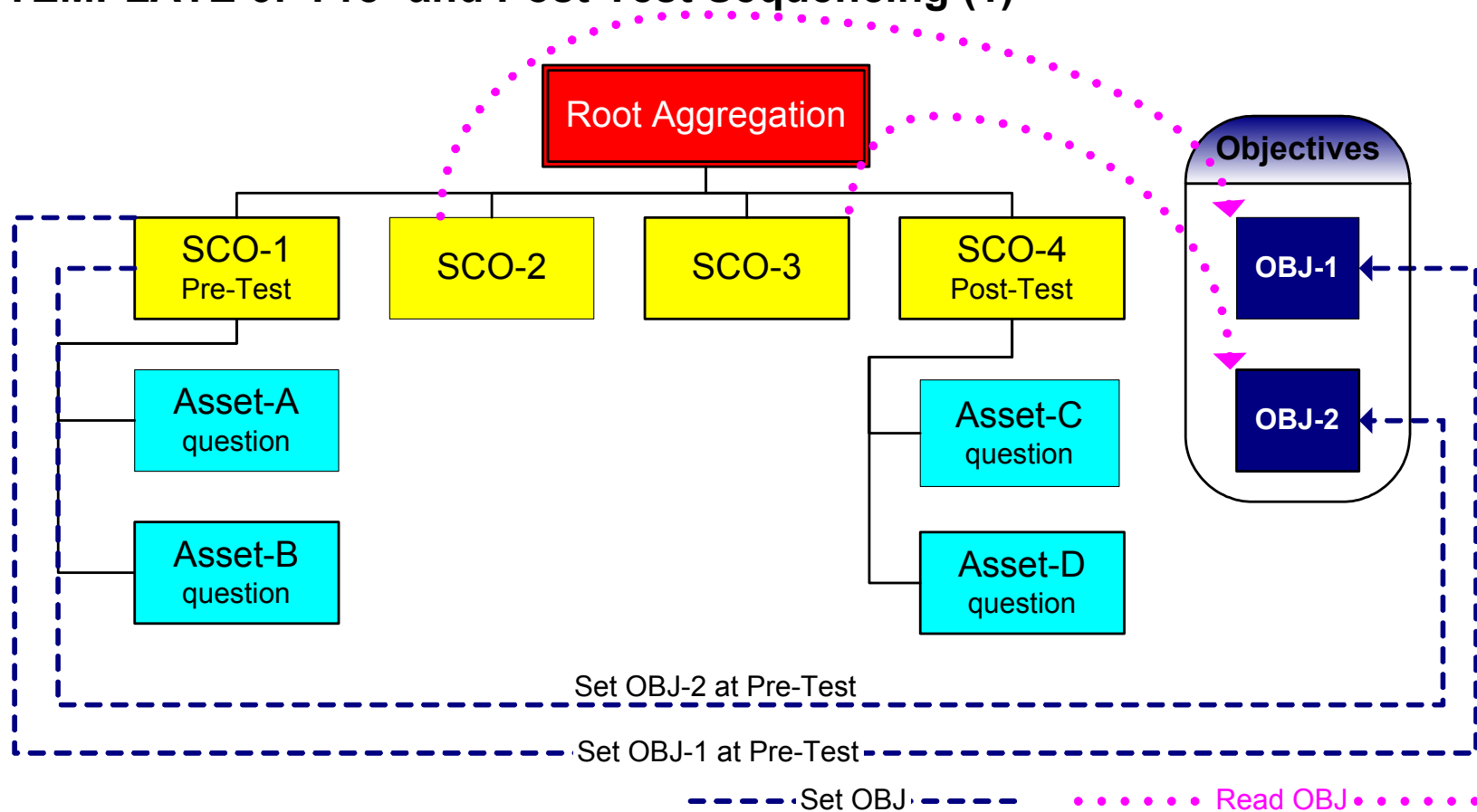| Behavior	SCORM Function
1. To complete the Root Aggregation, the learner must pass the post-test in SCO-3.	Root Aggregation: All: satisfied, completed SCO-1: isRolledup=false SCO-2: isRolledup=false SCO-3: isRolledup=true
2. The learner must complete SCO-1 before attempting SCO-2. The learner must complete SCO-2 before attempting SCO-3.	Root Aggregation: Flow=true; Choice=false
3. To complete SCO-3, both objectives must be passed.	*No unique SCORM function*
4. If the learner fails OBJ-1 in SCO-3, then present SCO-1.	SCO-3: set OBJ-1 SCO-1: skip if OBJ-1 passed
5. If the learner fails OBJ-2 in SCO-3, then present SCO-2.	SCO-3: set OBJ-2 SCO-2: skip if OBJ-2 passed
6. Allow two attempts for SCO-1, SCO-2, and SCO-3.	SCO-1, SCO-2, SCO-3: Attempt Limit=2
7. If the learner fails SCO-3 on attempt 2, the learner is halted in training and requires manual intervention.	*No unique SCORM function*

Template 5 Rules *(Application B)*:

Behavior	SCORM Function
1. To complete the Root Aggregation, the learner must pass the post-test in SCO-3.	Root Aggregation: All: satisfied, completed SCO-1: isRolledup=false SCO-2: isRolledup=false SCO-3: isRolledup=true
2. The learner can complete the SCOs in any order.	Root Aggregation: Flow=true; Choice=true
3. If the learner fails OBJ-1 in SCO-3, then present SCO-1.	SCO-3: set OBJ-1 SCO-3: skip if OBJ-1 passed
4. If the learner fails OBJ-2 in SCO-3, then present SCO-2.	SCO-3: set OBJ-2 SCO-2: skip if OBJ-2 passed
5. Allow as many attempts as needed to complete SCO-3.	*No unique SCORM function*

TEMPLATE 6: Pre- and Post-Test Sequencing (1)

Root Aggregation
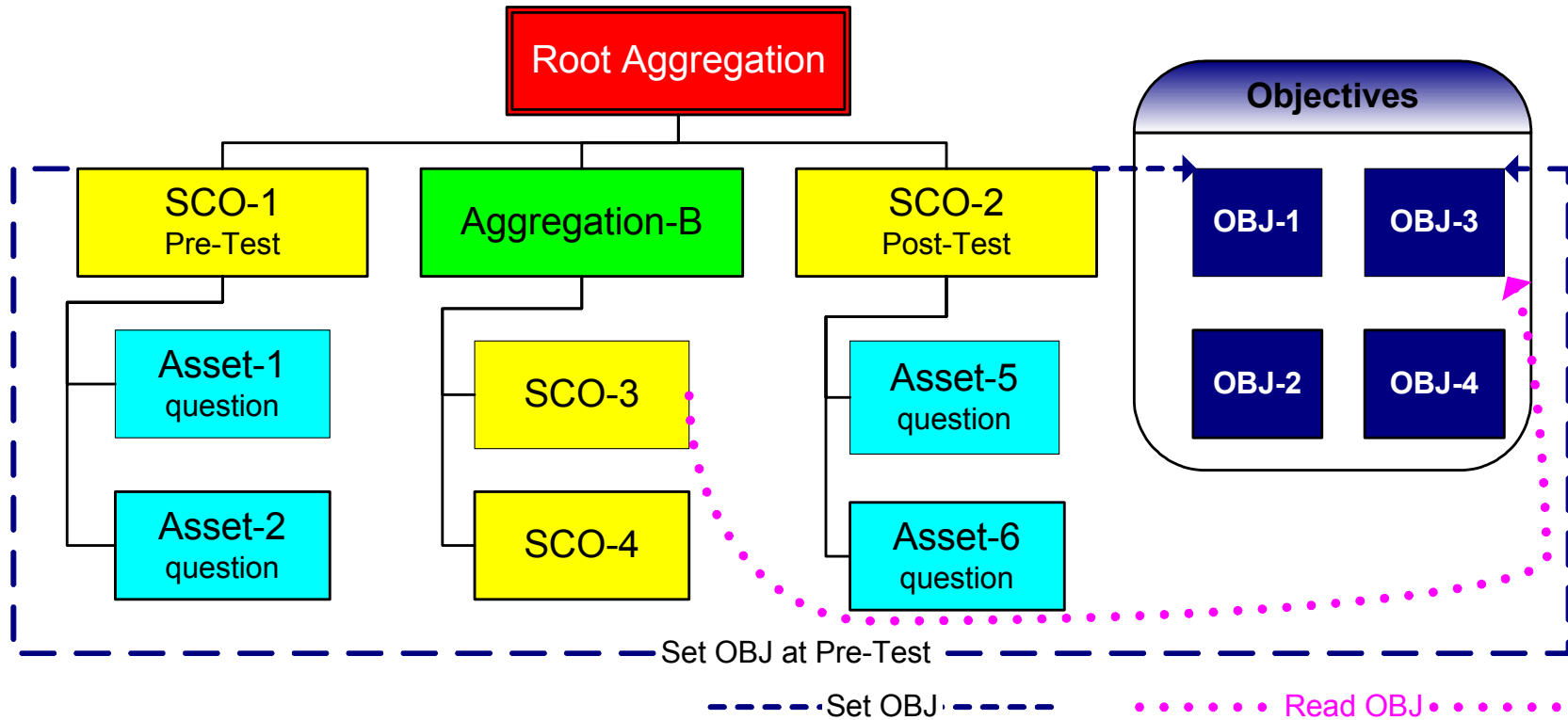
| SCO-1 Pre-Test | SCO-2 | SCO-3 | SCO-4 Post-Test |

Asset-A question

Asset-B question

Asset-C question

Asset-D question

Objectives

OBJ-1

OBJ-2

Set OBJ-2 at Pre-Test

Set OBJ-1 at Pre-Test

- - - Set OBJ - - -   · · · · Read OBJ · · · ·

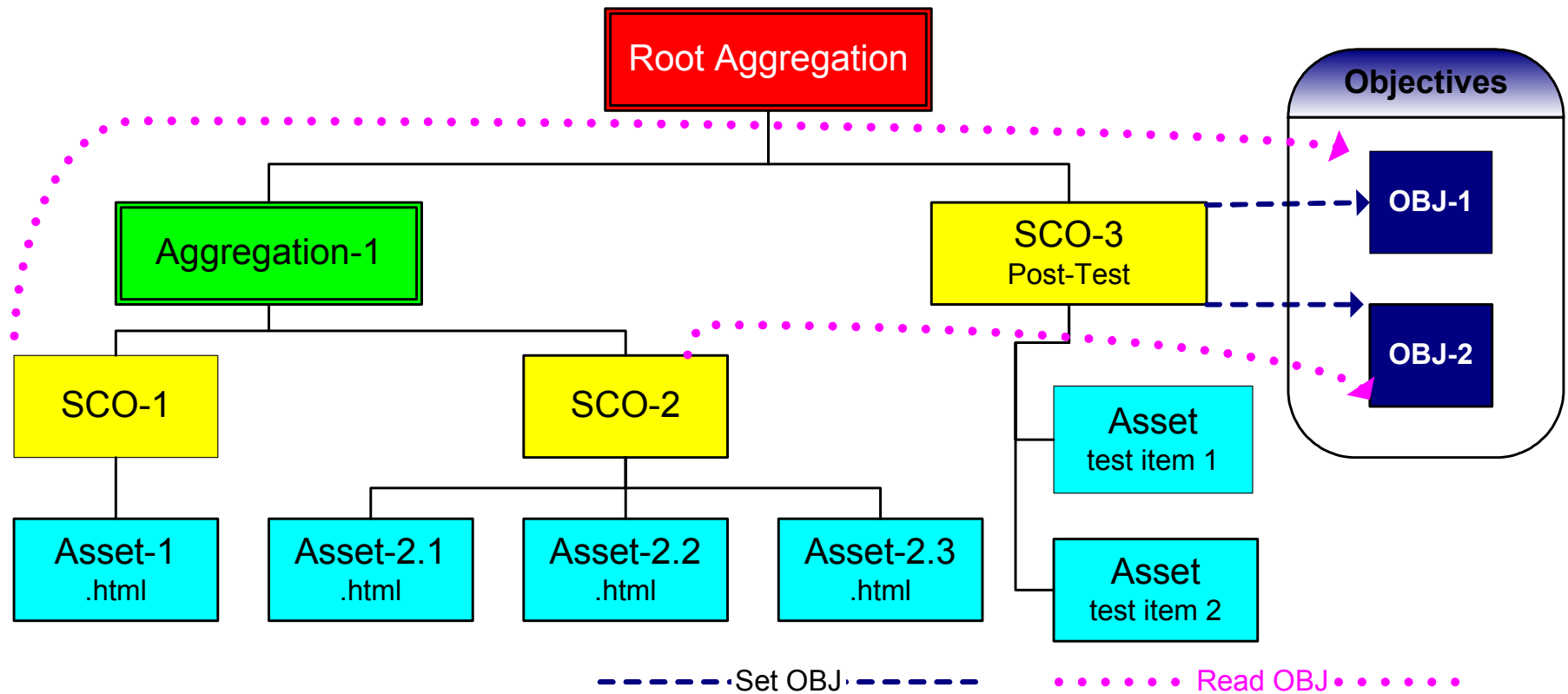| Template 6 Rules:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must pass the post-test in SCO-4.	Root Aggregation: All: satisfied, completed SCO-1: isRolledup=false SCO-2: isRolledup=false SCO-3: isRolledup=false SCO-4: isRolledup=true
2. The learner must complete the pre-test in SCO-1 before attempting SCO-2 or SCO-3.	Root Aggregation: Flow=true; Choice=false
3. The learner can return to SCO-1 from SCO-2 at any time.	Root Aggregation: Forward Only=false
4. If the learner fails OBJ-1 in SCO-1, then present SCO-2.	SCO-1: set OBJ-1 SCO-2: skip if OBJ-1 passed
5. If the learner fails OBJ-2 in SCO-1, then present SCO-3.	SCO-1: set OBJ-2 SCO-3: skip if OBJ-2 passed
6. To complete SCO-4, both test items must be passed.	*No unique SCORM function*
7. If the learner fails SCO-4, then the learner is halted in training and requires manual intervention.	*No unique SCORM function*

TEMPLATE 7: Pre- and Post-Test Sequencing (2)

NOTE: Not all links between SCOs and objectives are shown here.

Template 7 Rules:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must pass the post-test in SCO-2.	Root Aggregation: All: satisfied, completed SCO-1: isRolledUp=false Aggregation-B: isRolledUp=false SCO-2: isRolledUp=true
2. The learner must complete the pre-test in SCO-1 before attempting Aggregation B or SCO-2.	SCO-1: If not complete, Deny Forward Progress
3. The learner can return to SCO-3 from SCO-4 at any time.	Aggregation-B: Forward Only=False; Flow=true; Choice=true
4. If the learner fails OBJ-3 in SCO-1, then present SCO-3.	SCO-1: set OBJ-3 SCO-3: skip if OBJ-3 passed
5. If the learner fails OBJ-4 in SCO-1, then present SCO-4.	SCO-1: set OBJ-4 SCO-4: skip if OBJ-4 passed
6. The learner cannot return to SCO-1 or SCO-2 once Aggregation-B is attempted.	Root Aggregation: Flow=true; Forward-Only=true; Choice=false
7. To complete SCO-2, OBJ-1 and OBJ-2 must be passed.	*No unique SCORM function*
8. If the learner fails OBJ-1 or OBJ-2, then the learner is halted in training and requires manual intervention.	*No unique SCORM function*

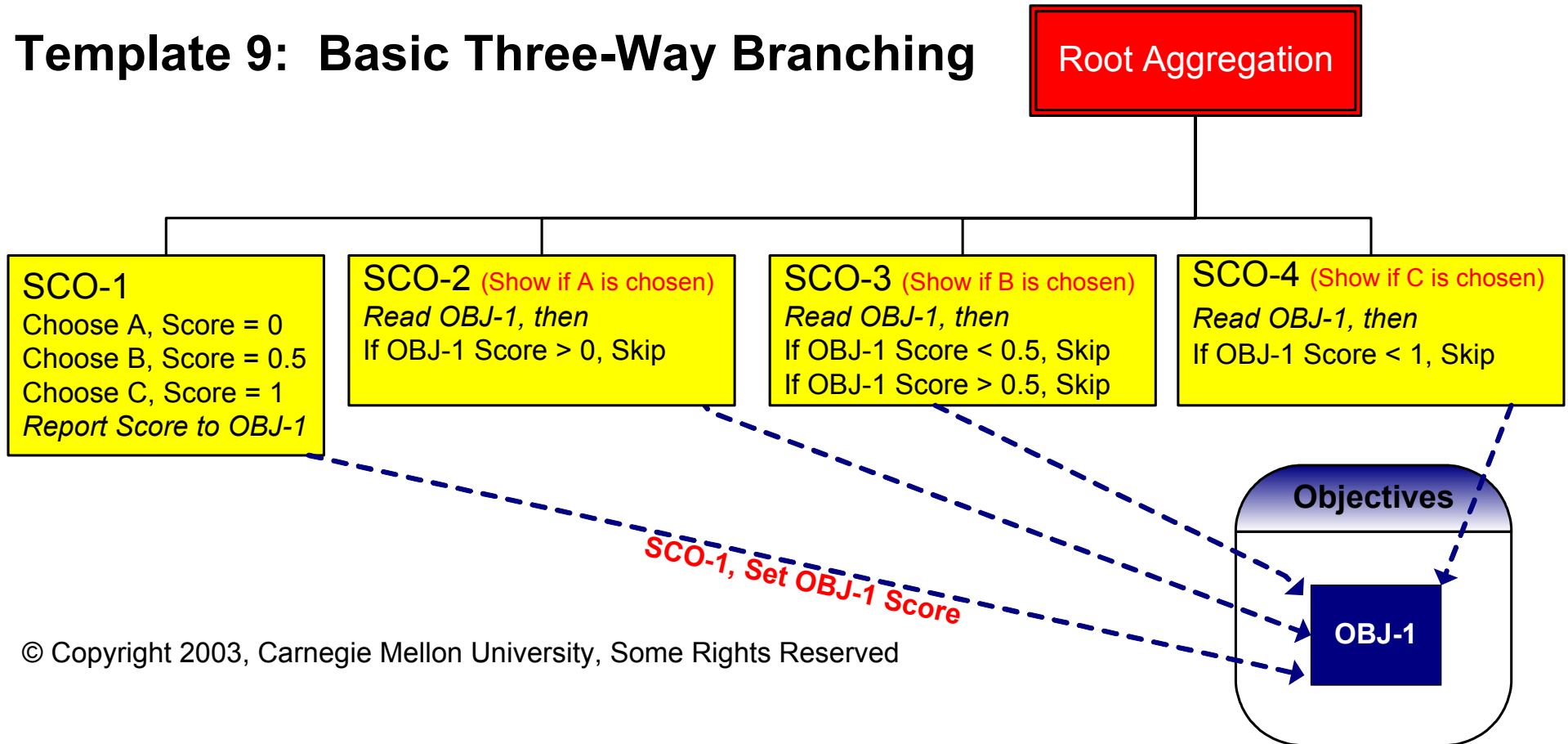## TEMPLATE 8: Remediating Using Objectives (2)

Set OBJ — — —   Read OBJ • • • • • •

© Copyright 2003, Carnegie Mellon University, Some Rights Reserved

Template 8 Rules:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must pass the post-test in SCO-3.	Root Aggregation: All satisfied, completed Aggregation-1: isRolledup=false SCO-3: isRolledup=true
2. The learner must complete Aggregation-1 before attempting SCO-3.	Root Aggregation: Flow=true; Deny Forward Progress=true
3. The learner can return to SCO-1 from SCO-2 at any time.	Aggregation-1: Forward Only=false
4. To complete SCO-3, both objectives must be passed.	*No unique SCORM function*
5. If the learner fails OBJ-1 in SCO-3, then present SCO-1.	SCO-3: set OBJ-1 SCO-1: skip if OBJ-1 passed
6. If the learner fails OBJ-2 in SCO-3, then present SCO-2.	SCO-3: set OBJ-2 SCO-2: skip if OBJ-2 passed
7. Allow two attempts for SCO-1, SCO-2, and SCO-3.	SCO-1, SCO-2, SCO-3: Attempt Limit=2
8. If the learner fails SCO-3 on the second attempt, then halt the learner in training and require manual intervention.	*No unique SCORM function*

Template 9: Basic Three-Way Branching

Root Aggregation

SCO-1
Choose A, Score = 0
Choose B, Score = 0.5
Choose C, Score = 1
Report Score to OBJ-1

SCO-2 (Show if A is chosen)
Read OBJ-1, then
If OBJ-1 Score > 0, Skip

SCO-3 (Show if B is chosen)
Read OBJ-1, then
If OBJ-1 Score < 0.5, Skip
If OBJ-1 Score > 0.5, Skip

SCO-4 (Show if C is chosen)
Read OBJ-1, then
If OBJ-1 Score < 1, Skip

Objectives

OBJ-1

SCO-1, Set OBJ-1 Score

Template 9 Rules *(Application A)*:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must pass two SCOs (SCO-1 and the *one* other SCO that is chosen by the sequencer). Rule 2 will ensure that SCO-1 is one of the two that is completed.	Root Aggregation: Completed if at least two children completed
2. The learner must do SCO-1 first.	Root Aggregation: Flow=true
3. Based on the learner's performance on the pre-test, branch to only one of the other three SCOs.	Root Aggregation: Choice=false SCO-1: set OBJ-1 SCO-2: skip if OBJ-1.score > 0 SCO-3: skip if OBJ-1.score < 0.5 or OBJ-1.score > 0.5 SCO-4: skip if OBJ-1.score < 1

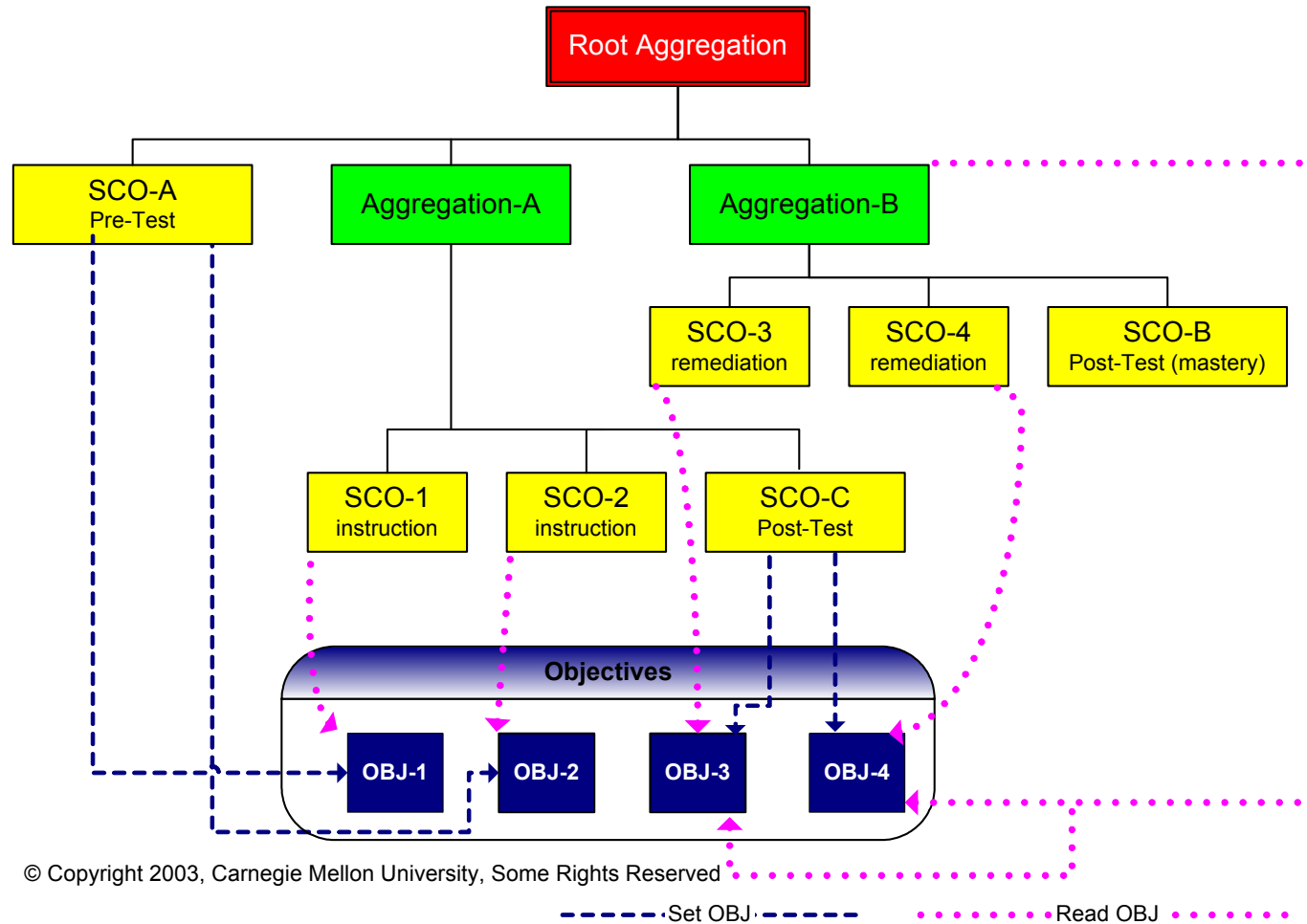| Template 9 Rules *(Application B)*:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must pass two SCOs (SCO-1 and the *one* other SCO that is chosen by the sequencer).	Root Aggregation: At least two completed, completed
2. The learner must do SCO-1 first.	Root Aggregation: Flow=true
3. Based on the learner's performance on the pre-test, branch to only one of the other three SCOs.	Root Aggregation: Choice=false SCO-1: set OBJ-2, OBJ-3, OBJ-4 SCO-2: skip if OBJ-2 passed SCO-3: skip if OBJ-3 passed SCO-4: skip if OBJ-4 passed

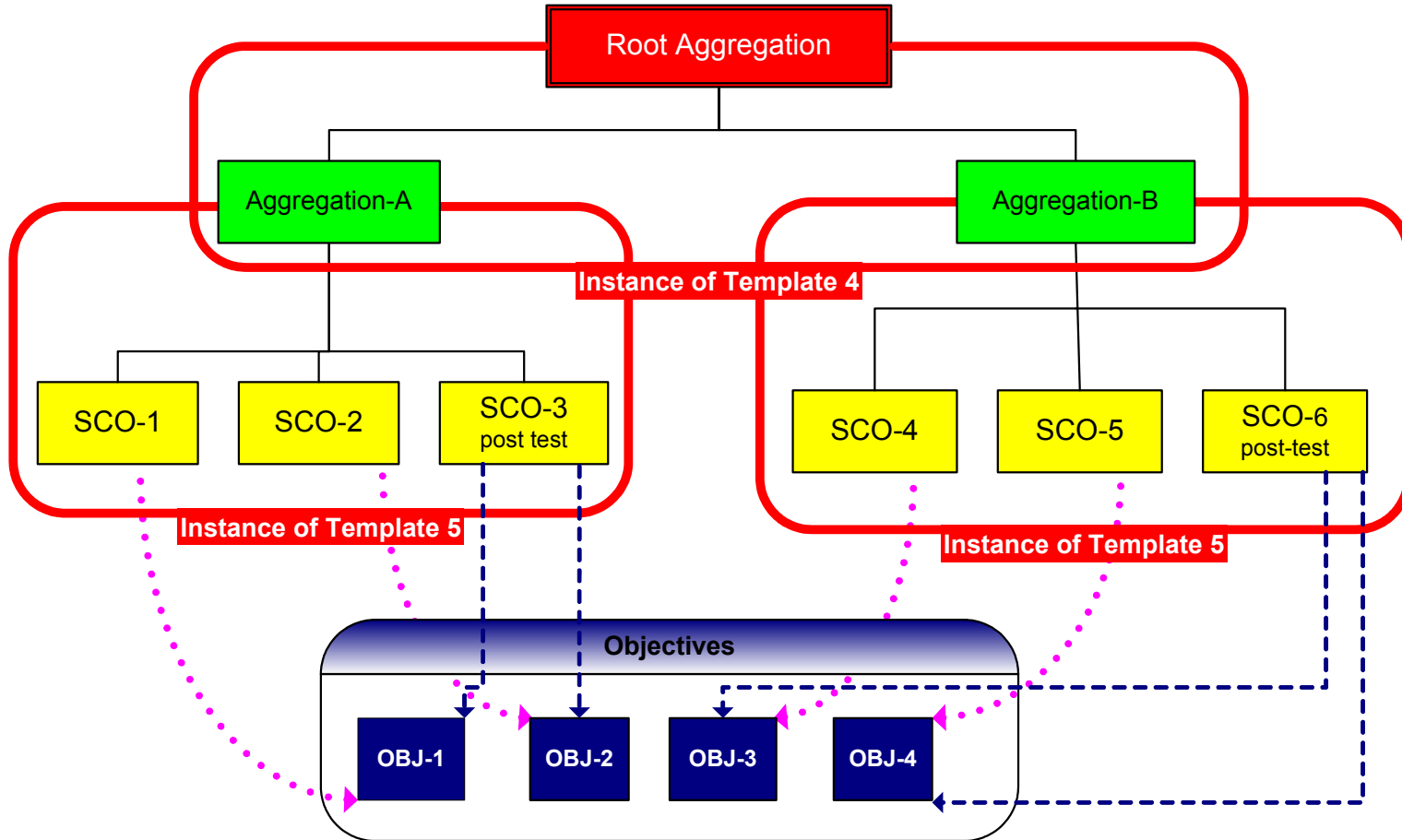**TEMPLATE 10: Pre- and Post-Test Sequencing With New Content for Remediation**

© Copyright 2003, Carnegie Mellon University, Some Rights Reserved

Set OBJ — — — — Read OBJ ● ● ● ● ● ●

SCORM Simple Sequencing Templates & Models

Template 10 Rules:	
Behavior	**SCORM Function**
1. To complete the Root Aggregation, the learner must pass the post-test in SCO-C **OR** the post-test in SCO-B.	Root Aggregation:Satisfied if one child satisfied SCO-A: isRolledup=false Aggregation-A: isRolledup=true Aggregation-B: isRolledup=true
2. The learner must complete the pre-test in SCO-A before attempting Aggregation-A. The learner cannot return to the Pre-Test from Aggregation-A.	Root Aggregation: Flow=true; Choice=false; Forward Only=true
3. If the learner fails OBJ-1 in SCO-A, then present SCO-1.	SCO-A: set OBJ-1 SCO-1: skip if OBJ-1 satisfied
4. If the learner fails OBJ-2 in SCO-A, then present SCO-2.	SCO-A: set OBJ-2 SCO-2: skip if OBJ-2 satisfied
5. The learner can return to SCO-1 from SCO-2 at any time.	Root Aggregation: Forward Only=false
6. To complete Aggregation-A, SCO-C must be passed.	SCO-1: isRolledup=false SCO-2: isRolledup=false Aggregation-A: rollup rule = All; satisfied, then satisfied; All: completed, then completed.
7. The learner will skip Aggregation-B if Aggregation-A is passed.	Aggregation-B: skip if OBJ-1 satisfied and OBJ-2 satisfied
8. If the learner fails OBJ-3 in SCO-C, then present SCO-3.	SCO-C: set OBJ-3 SCO-3: skip if OBJ-3 satisfied
9. If the learner fails OBJ-4 in SCO-C, then present SCO-4.	SCO-C: set OBJ-4 SCO-4: skip if OBJ-4 satisfied
10. If the learner fails SCO-B, then the learner is halted in training and requires manual intervention.	*No unique SCORM function*

MODEL 1: Remediating Multiple Aggregations

Set OBJ   Read OBJ

SCORM Simple Sequencing Templates & Models

Model 1 Rules *(Application A)*:		
Behavior	**SCORM Function**	**From Template**
1. To complete the Root Aggregation, the learner must complete Aggregation-A and Aggregation-B.	Root Aggregation: All: satisfied, completed.	4 (A)
2. The learner cannot start Aggregation-B until Aggregation-A is complete.	Aggregation-A: If Not Complete, Deny Forward Progress	4 (A)
3. To complete Aggregation-A, the learner must pass the post-test in SCO-3.	Aggregation-A: All: satisfied, completed SCO-1: isRolledUp=false SCO-2: isRolledUp=false SCO-3: isRolledUp=true	5 (A)
4. The learner must complete SCO-1 before attempting SCO-2. The learner must complete SCO-2 before attempting SCO-3.	Aggregation-A: Flow=true; Choice=false	5 (A)
5. To complete SCO-3, both objectives must be passed.	*No unique SCORM function*	5 (A)
6. If the learner fails OBJ-1 in SCO-3, then present SCO-1.	SCO-3: set OBJ-1 SCO-1: skip if OBJ-1 passed	5 (A)
7. If the learner fails OBJ-2 in SCO-3, then present SCO-2.	SCO-3: set OBJ-2 SCO-2: skip if OBJ-2 passed	5 (A)
8. Allow two attempts for SCO-1, SCO-2, and SCO-3.	SCO-1, SCO-2, SCO-3: Attempt Limit=2	5 A)
9. If the learner fails SCO-3 on attempt 2, the learner is halted in training and requires manual intervention.	*No unique SCORM function*	5 (A)
10. To complete Aggregation-B, the learner must pass the post-test in SCO-6.	Aggregation-B: All: satisfied, completed SCO-4: isRolledUp=false SCO-5: isRolledUp=false SCO-6: isRolledUp=true	5 (A)

Model 1 Rules *(Application A con't)*:		
11. The learner must complete SCO-4 before attempting SCO-5. The learner must complete SCO-5 before attempting SCO-6.	Aggregation-B: Flow=true; Choice=false	5 (A)
12. To complete SCO-6, both objectives must be passed.	*No unique SCORM function*	5 (A)
13. If the learner fails OBJ-3 in SCO-6, then present SCO-4.	SCO-6: set OBJ-3 SCO-4: skip if OBJ-3 passed	5 (A)
14. If the learner fails OBJ-4 in SCO-6, then present SCO-5.	SCO-6: set OBJ-4 SCO-5: skip if OBJ-4 passed	5 (A)
15. Allow two attempts for SCO-4, SCO-5, and SCO-6.	SCO-4, SCO-5, SCO-6: Attempt Limit=2	5 (A)
16. If the learner fails SCO-6 on attempt 2, the learner is halted in training and requires manual intervention.	*No unique SCORM function*	5 (A)

SCORM Simple Sequencing Templates & Models

Model 1 Rules *(Application B)*:		
Behavior	**SCORM Function**	**From Template**
1. To complete the Root Aggregation, the learner must complete Aggregation-A and Aggregation-B.	Root Aggregation: Rollup: All: completed, completed.	4 (B)
2. To complete each Aggregation, the learner must complete the post-tests within the Aggregations. (See rules 4 and 9).	*No SCORM function*	4 (B)
3. The learner can view the Aggregations in any order.	Root Aggregation: Flow and Choice=true	4 (B)
4. To complete Aggregation-A, the learner must pass the post-test in SCO-3.	Aggregation-A: All: satisfied, completed SCO-1: isRolledup=false SCO-2: isRolledup=false SCO-3: isRolledup=true	5 (B)
5. The learner can complete the SCOs in any order.	Aggregation-A: Flow=true; Choice=true	5 (B)
6. If the learner fails OBJ-1 in SCO-3, then present SCO-1.	SCO-3: set OBJ-1 SCO-1: skip if OBJ-1 passed	5 (B)
7. If the learner fails OBJ-2 in SCO-3, then present SCO-2.	SCO-3: set OBJ-2 SCO-2: skip if OBJ-2 passed	5 (B)
8. Allow as many attempts as needed to complete SCO-3.	*No unique SCORM function*	5 (B)
9. To complete Aggregation-B, the learner must pass the post-test in SCO-6.	Aggregation-B: All: satisfied, completed SCO-4: isRolledup=false SCO-5: isRolledup=false SCO-6: isRolledup=true	5 (A)

| Model 1 Rules *(Application B con't)*:		
10. The learner must complete SCO-4 before attempting SCO-5. The learner must complete SCO-5 before attempting SCO-6.	Root Aggregation: Flow=true; Choice=false	5 (A)
11. To complete SCO-6, both objectives must be passed.	*No unique SCORM function*	5 (A)
12. If the learner fails OBJ-3 in SCO-6, then present SCO-4.	SCO-6: set OBJ-3 SCO-4: skip if OBJ-3 passed	5 (A)
13. If the learner fails OBJ-4 in SCO-6, then present SCO-5.	SCO-6: set OBJ-4 SCO-5: skip if OBJ-4 passed	5 (A)
14. Allow two attempts for SCO-4, SCO-5, and SCO-6.	SCO-4, SCO-5, SCO-6: Attempt Limit=2	5 (A)
15. If the learner fails SCO-6 on attempt 2, the learner is halted in training and requires manual intervention.	*No unique SCORM function*	5 (A)

This Page Intentionally Left Blank

MODEL 2: Mastery Testing Multiple Aggregations
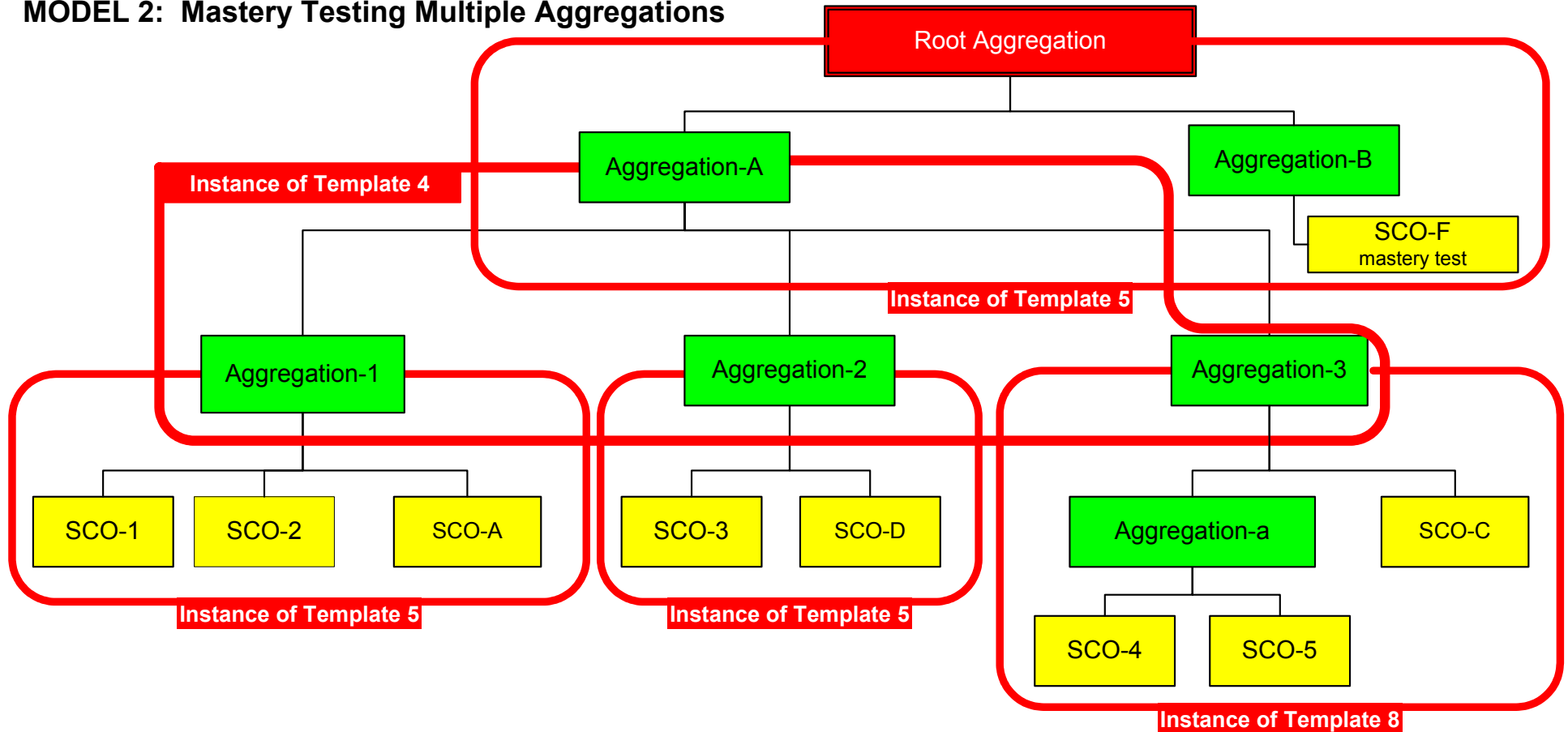
© Copyright 2003, Carnegie Mellon University, Some Rights Reserved

NOTE: Links to objectives within Aggregations 1, 2, and 3 are not shown. Refer to Templates 5 and 8 for specific details.

Model 2 Rules:		
Behavior	**SCORM Function**	**From Template**
1. To complete the Root Aggregation, the learner must pass the mastery test (SCO-F) in Aggregation-B.	All satisfied, completed Aggregation-A: isRolledup=false Aggregation-B: isRolledup=true	5 (A)
2. The learner must complete Aggregation-A before attempting Aggregation-B.	Flow=true; Deny Forward Progress=true	5 (A)
3. To complete Aggregation-A, the learner must complete Aggregation-1, Aggregation-2, and Aggregation-3 in order.	Flow=true; Deny Forward Progress=true	4 (A)
4. To complete Aggregation-1, the learner must pass the post-test in SCO-A.	All satisfied, completed SCO-1: isRolledup=false SCO-2: isRolledup=false SCO-A: isRolledup=true	5 (A)
5. The learner must complete SCO-1 before attempting SCO-2. The learner must complete SCO-2 before attempting SCO-A.	Flow = true; Deny Forward Progress=true	5 (A)
6. The learner can return to SCO-1 from SCO-2 at any time.	Aggregation-1: Forward Only=false	5 (A)
7. The learner cannot return to SCO-1 or SCO-2 once Aggregation-a is attempted.	Aggregation-1: Forward Only=true	5 (A)
8. If the learner fails OBJ-1 in SCO-A, then present SCO-1.	SCO-A: set OBJ-1 SCO-1: skip if OBJ-1 passed	5 (A)
9. If the learner fails OBJ-2 in SCO-A, then present SCO-2.	SCO-A: set OBJ-2 SCO-2: skip if OBJ-2 passed	5 (A)
10. Allow two attempts for SCO-1, SCO-2, and SCO-A.	SCO-1, SCO-2, SCO-A: Attempt Limit=2	5 (A)
11. If the learner fails SCO-A on attempt 2, the learner is halted in training and requires manual intervention.	*No unique SCORM function*	5 (A)

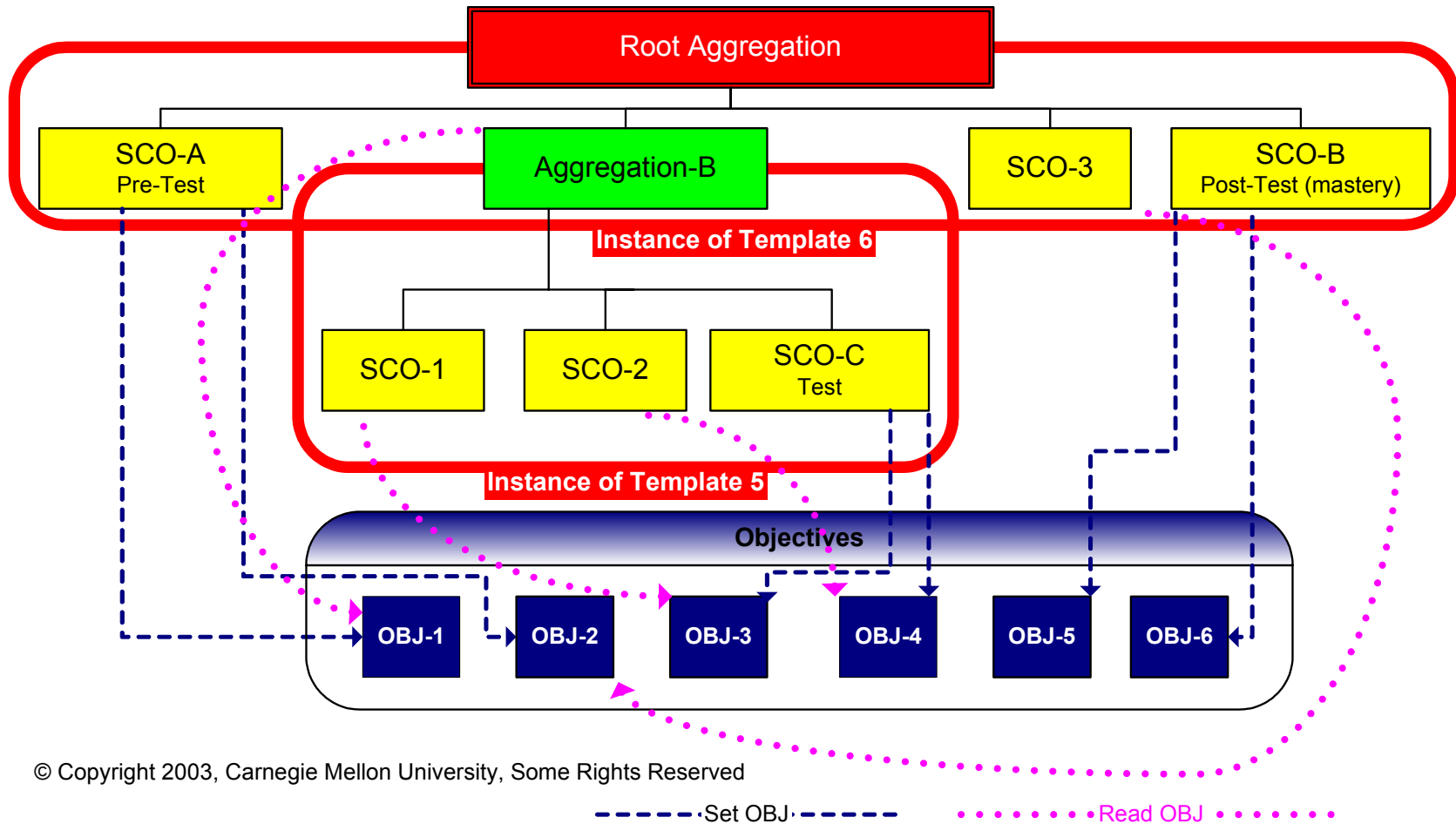| Model 2 Rules: (con't)		
12. To complete Aggregation-2, the learner must pass the post-test in SCO-D.	All satisfied, completed SCO-3: isRolledup=false SCO-D: isRolledup=true	5 (A)
13. The learner must complete SCO-3 before attempting SCO-D.	Flow=true; Deny Forward Progress=true	5 (A)
14. The learner cannot return to SCO-3 once SCO-D is attempted.	Aggregation-2: Forward Only=True	5 (A)
15. If the learner fails OBJ-3 in SCO-D, then present SCO-3.	SCO-D: set OBJ-3 SCO-3: skip if OBJ-3 passed	5 (A)
16. Allow two attempts for SCO-3 and SCO-D.	SCO-3, SCO-D: Attempt Limit=2	5 (A)
17. If the learner fails SCO-D on attempt 2, the learner is halted in training and requires manual intervention.	*No unique SCORM function*	5 (A)
18. To complete Aggregation-3, the learner must pass the post-test in SCO-C.	Aggregation-3: All satisfied, completed Aggregation-a: isRolledup=false SCO-C: isRolledup=true	8
19. The learner must complete Aggregation-a before attempting SCO-C.	Aggregation-3: Flow; Deny Forward Progress=True	8
20. The learner can return to SCO-4 from SCO-5 at any time.	Aggregation-a: Forward Only=false	8
21. If the learner fails OBJ-4 in SCO-C, then present SCO-4.	SCO-C: set OBJ-4 SCO-4: skip if OBJ-4 passed	8
22. If the learner fails OBJ-5 in SCO-C, then present SCO-5.	SCO-C: set OBJ-5 SCO-5: skip if OBJ-5 passed	8
23. Allow two attempts for SCO-4, SCO-5, and SCO-C.	SCO-4, SCO-5, SCO-C Attempt Limit=2	8
24. If the learner fails SCO-C on attempt 2, the learner is halted in training and requires manual intervention.	*No unique SCORM function*	8

This Page Intentionally Left Blank

MODEL 3: Pre- and Post-Test Sequencing with Aggregations

Set OBJ - - - - - - Read OBJ ··········

Model 3 Rules:		
Behavior	**SCORM Function**	**From Template**
1. To complete the Root Aggregation, the learner must pass the post-test in SCO-B.	Root Aggregation: All: satisfied, completed SCO-A: isRolledup=false Aggregation-B: isRolledup=false SCO-3: isRolledup=false SCO-B: isRolledup=true	6
2. The learner must complete the pre-test in SCO-A before attempting Aggregation-B or SCO-3.	Root Aggregation: Flow=true; Choice=false	6
3. The learner can return to SCO-A from Aggregation-B or SCO-3 at any time.	Root Aggregation: Forward Only=false	6
4. If the learner fails OBJ-1 in SCO-A, then present Aggregation-B.	SCO-A: set OBJ-1 Aggregation-B: skip if OBJ-1 passed	6
5. If the learner fails OBJ-2 in SCO-A, then present SCO-3.	SCO-A: set OBJ-2 SCO-3: skip if OBJ-2 passed	6
6. To complete Aggregation-B, the learner must pass the post-test in SCO-C.	Aggregation-B: All: satisfied, completed SCO-1: isRolledup=false SCO-2: isRolledup=false SCO-C: isRolledup=true	5 (A)
7. The learner must complete SCO-1 before attempting SCO-2. The learner must complete SCO-2 before attempting SCO-3.	Aggregation-B: Flow=true; Choice=false	5 (A)
8. To complete SCO-C, both objectives must be passed.	*No unique SCORM function*	5 (A)
9. If the learner fails OBJ-3 in SCO-C, then present SCO-1.	SCO-C: set OBJ-3 SCO-1: skip if OBJ-3 passed	5 (A)
10. If the learner fails OBJ-4 in SCO-C, then present SCO-2.	SCO-C: set OBJ-4 SCO-2: skip if OBJ-4 passed	5 (A)
11. Allow two attempts for SCO-1, SCO-2, and SCO-C.	SCO-1, SCO-2, SCO-C: Attempt Limit=2	5 (A)
12. If the learner fails SCO-C on attempt 2, the learner is halted in training and requires manual intervention.	*No unique SCORM function*	5 (A)
13. If the learner fails SCO-B, then the learner is halted in training and requires manual intervention.	*No unique SCORM function*	6

Model 4: Traditional CBT Branching with Multiple Decisions

Root Aggregation

SCO-1
Choose A, Score = 0
Choose B, Score = 0.5
Choose C, Score = 1
Report Score

Aggregation-1
If OBJ-1 Score = 0, Skip
If OBJ-1 Score = 0.5, Skip
If OBJ-1 Score = 1, OK

Aggregation-2
If OBJ-1 Score = 0, Skip
If OBJ-1 Score = 0.5, OK
If OBJ-1 Score = 1, Skip

Aggregation-3
If OBJ-1 Score = 0, OK
If OBJ-1 Score = 0.5, Skip
If OBJ-1 Score = 1, Skip

SCO-2
Repeat SCO-1 rules
with OBJ-2

Aggregation-4
follow purple rules

Aggregation-5
If OBJ-2 Score = 0, Skip
If OBJ-2 Score = 0.5, OK
If OBJ-2 Score = 1, Skip

Aggregation-6
follow green rules

Objectives

OBJ-1

OBJ-2

OBJ-3

SCO-1, Set OBJ-1

SCO-5, Set OBJ-2

SCO-3
Repeat SCO-1 rules
with OBJ-3

Aggregation-7
follow purple rules

Aggregation-8
follow teal rules

Aggregation-9
If OBJ-3 Score = 0, OK
If OBJ-3 Score = 0.5, Skip
If OBJ-3 Score = 1, Skip

SCO-9, Set OBJ-3

© Copyright 2003, Carnegie Mellon University, Some Rights Reserved

Shows one possible path learner takes through organization

NOTE: In this model, the learner always sees a yellow SCO at each
layer in the tree because content can only be contained in a
SCO and not in an aggregation.

This Page Intentionally Left Blank