Authoring OneApp Learn Content

Contents

[About this document 3](#_Toc236796309)

[What is OneApp Learn? 4](#_Toc236796310)

[Learning structure 4](#_Toc236796311)

[Overview of Authoring 5](#_Toc236796312)

[Authoring Workflow 5](#_Toc236796313)

[Quick Start 5](#_Toc236796314)

[Authoring Tools 8](#_Toc236796315)

[Functions of the Content Structures 8](#_Toc236796316)

[Minimum authoring requirements for content providers 9](#_Toc236796317)

[Supplemental content for content providers 9](#_Toc236796318)

[Display Screens and Targeted Sizes 10](#_Toc236796319)

[Content Changes/Updates 10](#_Toc236796320)

[The Microsoft Infopath Forms for OneApp Learn Authoring 11](#_Toc236796321)

[Overview 11](#_Toc236796322)

[List of the InfoPath Forms 12](#_Toc236796323)

[The Basics of OneApp Learn Form Authoring 13](#_Toc236796324)

[Text formatting is not supported 13](#_Toc236796325)

[Line Breaks 13](#_Toc236796326)

[Lists 13](#_Toc236796327)

[Short Titles 13](#_Toc236796328)

[IDs and IDREFs 13](#_Toc236796329)

[Filepaths 13](#_Toc236796330)

[Filenames 14](#_Toc236796331)

[Adding/Removing Items in general 14](#_Toc236796332)

[Removing, Ordering and Cut and Paste 14](#_Toc236796333)

[Handling Images 14](#_Toc236796334)

[Saving and Naming your files 15](#_Toc236796335)

[Details of the Document Types 16](#_Toc236796336)

[SubjectAreas 17](#_Toc236796337)

[Provider Authoring 17](#_Toc236796338)

[A Note about Inner-Topic Navigation (Activity to Activity navigation) 19](#_Toc236796339)

[Location Notification 19](#_Toc236796340)

[ContentAssets 20](#_Toc236796341)

[Note about content for mobile devices 20](#_Toc236796342)

[Content Outline 22](#_Toc236796343)

[Text and Picture Content 23](#_Toc236796344)

[Flashcard Game Content 28](#_Toc236796345)

[Adventure Game Content 31](#_Toc236796346)

[Sort Game Content 33](#_Toc236796347)

[Pan Image Content 34](#_Toc236796348)

[Product Details 36](#_Toc236796349)

[Location Details 40](#_Toc236796350)

[Location Town View 40](#_Toc236796351)

[Location Results View 40](#_Toc236796352)

[Location Detail View 40](#_Toc236796353)

[Location Notification 40](#_Toc236796354)

[Forms and uses of LocationDetails.xml documents 40](#_Toc236796355)

[Assessment Content ( Test ) 42](#_Toc236796356)

[Tip Content 45](#_Toc236796357)

[Offer Details 47](#_Toc236796358)

# About this document

This document describes the processes for authoring for the Microsoft OneApp Learn system.

These processes include considering, and creating materials such as product details, assessments, mobile-ready course content, training location information, and DVD content ordering information.

# What is OneApp Learn?

The OneApp Learn solution is comprised of the following pieces:

* An instructional framework for improving knowledge, comprehension, and recall of education material available to your mobile phone. The framework consists of many types of engaging activities, tests, and peer collaboration.
* A mobile phone application consisting of an integrated course catalog system, a training facilities locator system, content presentation and interact ivies, and testing monitoring systems.
* Services for internet and phone networks for publishing and provisioning training content.
* An XML authoring kit for creating content and metadata for the system.

## Learning structure

The following depicts the overall high level structure across OneApp Learn

# Overview of Authoring

The content provided to the system is made up of XML and image files. The XML files are segmented heavily to enable performance for the mobile networks, as well as allowing authors to fine tune the amount of content displaying on potentially small screens.

At this time, authoring entails:

Using your favorite XML editor to create XML content instances which define your product

Managing the XML files and their inter-relationships

Reviewing your content (quality control)

Handing off the content to the OneApp Learn service partner

These processes include considering, and creating materials such as product details, assessments, mobile-ready course content, training location information, and DVD content ordering information.

## Authoring Workflow

As one document may inform the other, the illustration below is a representation of the recommended flow for content production.

### Quick Start

These steps use the sample project Computer Basics as the project template.

#### Prep a new Project

Copy the Computer Basics sample project from the Authoring Kit into your own project folder.

#### 1) Edit the subjectareas.xml document

#### This document relates your product to the course catalog. It uses the same content model as the full entire course catalog structure but we only need to related a single product to a catalog subject area.

a.) Edit the Subjects section to match your contents subject area. The Subject ID and Subject Name should be accurate, If you don't know which subject ID to use, content your OneApp Learn service support provider.

b.) In the Product Results section, change the only row so that ***result title*** and ***result context*** reflect your products content

#### 2.) Add/Modify new content and edit contentassets.xml, contentoutline.xml

a.) Author content using one of the content type form templates.

TextAndPicture

SortGame

FlashcardGame

PanImage

\* Filenames should be lowercase.

\* For image, insure they are PNG and contain the PNG file extension ***.png.*** The filenames should be lower case whenever possible.

\* Use an easy to remember file naming convention:

***[Activity Number][Activiy Type Abbr][optional iteration number as needed)]***

A single topic might contain 4 reading activities ( 4 files )

**01intro.xml**, would be topic01's TextAndPicture Intro activity

**01cont.xml**, would be topic01's TextAndPicture Continuation activity

**01capimg1.xml**, would be a TextAndPicture CaptionedImage activity

**01capimg2.xml**, would be a TextAndPicture another CaptionedImage activity

\* Take note of PNG image filenames you add to the content templates:

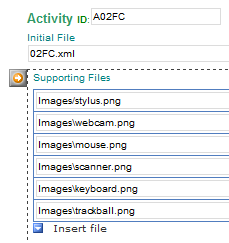
b.) Edit the ***contentassets.xml*** file to insure they reflect the content files and dependencies.

i.) Use an easy to remember activity ID convention:

A + [***filename -*** minus the file extension]

The above topic would contain the following Activity IDs:

**A01intro,** A**01cont, A01capimg1, and A01capimg2**ii.) Insure an related images to the activities files are noted within the supporting files section for that activity. For simplicity here, these file should exist within the images file folder of your project.



C.) Edit the ***contentoutline.xml*** file to insure they correct activities are targets as the content topics.

i.) Use an easy to remember topic ID convention:

T + [topic number]

**T01** would be related to Activity ID **A01intro**

**T02** would be related to Activity ID **A02intro**

#### 3.) Modify the assessment.xml

a.) Author content using one of the content type form templates.

TextAndPicture

#### 4.) Modify the productdetails.xml

a.) Modify product detail information such as title, byline, outline, and details.

b.) If you product has a DVD component add DVD component SKU number

c.) If your product has a Program Access components, provide the access codes

#### Prep for Handoff

Copy all the files , sub folders and all, into a zip file to prepare it for handoff to the service provider. Name the zip the same or similar title to your product title.

# Authoring Tools

The following are being provided as part of this authoring toolkit.

* This document which serves as a guide for authoring.
* Infopath forms are used to author the content documents.
* Common Image assets
* XML samples and XSD for schemas used for validation with your own XML authoring systems

## Functions of the Content Structures

#### Product Details

This provides the overall detail page content for the catalog system including Title, Byline, Outline, Detail, and product components list including content, DVD SKU, and location references.

#### SubjectAreas

This defines the subject area, product relationships. Used for catalogue navigation.

#### ContentAssets

This defines the files used as *activities* within the system, and the dependencies files of an activity such as related images.

#### ContentOutline

This defines the course topics and the relationship between the topic, and it’s first activity.

#### AssessmentContent

Defines an assessment which targets course topic content.

#### Tips

Define tips of the day for your product.

#### Activity Content Types (Text and Picture, PanImage, Flashcard, Adventure, and Sort)

These types provide the learning activities such as identify, recognize, memorize, select, and locate. The content types within include flashcard, sorting, adventure games, as well as variant forms of reading and image topic types.

#### LocationDetails

This provides location information such as Town/Region information, optional location result structures, and full location provider details structures.

#### OfferDetails

Define program access offers products can participate in.

## Minimum authoring requirements for content providers

The following are required for publishing to the OneApp Learn Service.

* A Product Details File
* Subject Areas File(s)
* Assessment Content File
* Course Outline File
* Content Assets File
* Any number of the following individual content activities as described within the Content Assets file
  + Text And Picture Content
  + Adventure Game Content
  + Flashcard Game Content
  + Pan Image Content
  + Sort Game Content

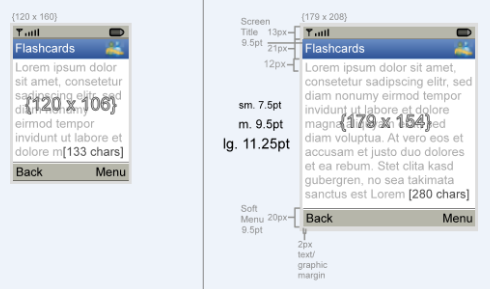
## Supplemental content for content providers

The following files need to be supplied or modified to provide location and DVD ordering information to the OneApp Learn System.

* Modify Product Details File
  + include references to locations
  + provide DVD SKU number
* Location Details File(s)

## Display Screens and Targeted Sizes

The following sizes, 3:4 120x160 and 3:4 179x208 screens sizes are being targeted.



Although it is recommended to avoid scrolling, across all content types, a single vertical scroll bar will be enabled and show if needed. For example, for reading content types such as Text and Picture, the entire reading area will be a focusable/scrollable control. The scroll should only appear if the content doesn’t fit within the view and again, authors should make every attempt to avoid scrolling content.

The content should be designed for the larger of these sizes (shown above), but fallback gracefully to the lower size.

Larger screens, such as 3:4, 240x320, are also supported as well, however, these sizes have much lower market saturation than the mid-size and small screens.

The content schema validation (XSD) should enable a majority of the graceful failure in content presentation. This is done by not allowing too much content on any single screen, and formatting images to only supporting image sizes. Note however, depending on letter space of specific words or combination of words, “graceful” may not always be possible on the smallest devices and content should be reviewed and changed to fit that smaller size as required.

### Content Changes/Updates

For any republishing of content to the OneApp Learn service provider, please contact that the service provider for specific instructions.

# The Microsoft Infopath Forms for OneApp Learn Authoring

## Overview

Using the Infopath forms are fairly straight forward. For any of the content types, there is an associated Infopath form (***.xsn***) within the Authoring Kit folder *Infopath Forms*. When you open a form by double clicking, you can then fill out the form in Infopath, and then save it as an XML file when you’re done. The resulting XML file is a OneApp Learn content file, and now and you can double click the XML file to open an edit it at any time later if you have changes to make. You follow the exact same process to create all of your content and metadata for your product.

## List of the InfoPath Forms

The following is the association between the types of content or metadata and the Infopath forms used to author that data.

|  |  |  |
| --- | --- | --- |
| Content Type | Sub Section or Activity | InfoPath Template |
| Product Details | Details | ProductDetails.xsn |
|  | Component, Content Outline Reference | ProductDetails.xsn |
|  | Component, DVD | ProductDetails.xsn |
|  | Component, Location References | ProductDetails.xsn |
| Subject Areas | Subjects | SubjectAreas.xsn |
|  | Related Products | SubjectAreas.xsn |
| Location Details | Location Details | LocationDetails.xsn |
|  | Town & Regions | LocationDetails.xsn |
|  | Location References | LocationDetails.xsn |
| Content Assets | Content Asset Items (Activity) | ContentAssets.xsn |
|  | File Groups | ContentAssets.xsn |
| Content Outline (Topics) |  | ContentOutline.xsn |
| Test/Assessment |  | AssessmentContent.xsn |
| Reading Text & Picture | Introduction | TextAndPicture.xsn |
|  | Continuation | TextAndPicture.xsn |
|  | Captioned Photo/Rendering | TextAndPicture.xsn |
|  | Summery | TextAndPicture.xsn |
| Panning line art viewing |  | PanImage.xsn |
| Flashcards |  | FlashcardGame.xsn |
| Sorting |  | SortGame.xsn |
| Tips of the Day |  | Tips.xsn |
| Offers |  | OfferDetails.xsn |

## 

## The Basics of OneApp Learn Form Authoring

The forms themselves contain some minor advise and instruction for authoring. The following is additional basic information about OneApp Learn InfoPath Form authoring.

### Text formatting is not supported

Presently, no inline formatting of text is supported. Such as bold, italic, underline, etc. If you need to provide emphasis to text, consider using upper case, quoting, or using parenthesis.

### Line Breaks

Specific form parts do support line breaks. "Text" elements support it line breaks and other forms of whitespace. Title, caption, or file path references do not support line breaks.

### Lists

Lists will only be possible in "Text" elements since there isn't yet a dedicated list style and these elements support line breaks. For indents and list bullets, consider leaving them out whenever possible. Optionally, for numbered lists, the form 1.) or 2) is commonly used. Likewise, for unordered lists, a "\*" is commonly used. For

### Short Titles

Some form fields will be very limiting in the amount of text you can enter. In this case it is recommended to provide abbreviations wherever possible, and use "..." to denote longer text information. Mainly fields like these are intended only as a reader guide.

### IDs and IDREFs

Throughout the forms, there will be field labeled as "ID:". These are identifier fields and they work somewhat like bookmarks. They are unique names given the parts of the document so that later they can be looked up easily. An IDREF stands for Identifier reference and it allows us to use and Identifier from one part of the document, inside another part of a document. These ID/IDREF relationships allow us to ties the parts of the documents together.

In many parts of the forms, Identifiers will be required. Take care with any whitespace and improper naming and use of **id** and **idref** attributes. If they are used for referencing and ID within another document, validated as a custom strings instead of a proper idref type ( no warning of an error will appear ). They are validated as an IDREF only when referencing an element within the same document.

### Filepaths

The value for system identifiers input should be relative, denoting a sibling, sibling/descendant relationship. For example, “*sharename*/*filename.ext*” and “topic10.xml” are valid.

### Filenames

Filenames should be lowercase.

Do not change the filenames for non-activity files such as contentassets.xml, contentoutline.xml, productdetails.xml, etc - keep these filenames the same.

For activity files, use an easy to remember file naming convention:

***[Activity Number][Activiy Type Abbr][optional iteration number as needed)]***

A single topic might contain 4 reading activities ( 4 files )

**01intro.xml**, would be topic01's TextAndPicture Intro activity

**01cont.xml**, would be topic01's TextAndPicture Continuation activity

**01capimg1.xml**, would be a TextAndPicture CaptionedImage activity

**01capimg2.xml**, would be a TextAndPicture another CaptionedImage activity

For image, insure they are PNG and contain the PNG file extension ***.png.*** The filenames should be lower case whenever possible.

### Adding/Removing Items in general

The  button allows you to choose one form part type or another one.

The  buttons allow access to inserting ***repeatable*** form parts.

The button allows access to inserting ***optional*** elements.

### Removing, Ordering and Cut and Paste

The  button allows removing a repeating form parts, inserting another part before or after the present part, and also enables cut and paste operations.

### Handling Images

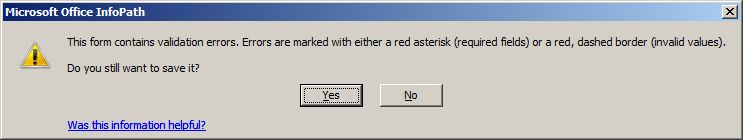
Some images are only authored as relative file path, however, some are meant to show up within the form to help better identify the content type as you author. When the picture icon is visable , click to select a picture. Insure the picture you select is within a relative directory to where you have saved or will save the form XML file your working on.

### Saving and Naming your files

Saving can be done through the InfoPath File/Save menu option. If the file hasn't been saved before, you'll be prompted to save it as an XML file, and to select a save destination.

#### Validation Errors

The following dialog may occur when trying to save your file:



This means the document is invalid and needs to be corrected before being published. This also contains instructions on how to identify and fix the errors.

#### File Naming

When naming the file, consider saving using file naming convention that works best for your production processes as well as for the OneApp Learn environment. To insure cross operating system support, insure your filenames do not include only the basic latin characters [A-z, 0-9] and don't use symbols of any kind. The file extension suffix (.xml), of course, is the exception here.

#### Standard Files

The following templates should be used to save a single file with a specific name:

ContentAssets.xsn: Save as *contentassets.xml*

ContentOutline.xsn: Save as *contentoutline.xml*

ProductDetails.xsn: Save as *productdetail.xml*

Assessment.xsn: Save as *assessment.xml*

# Details of the Document Types

The following sections iterate the details and authoring consideration for each topic type.

## SubjectAreas

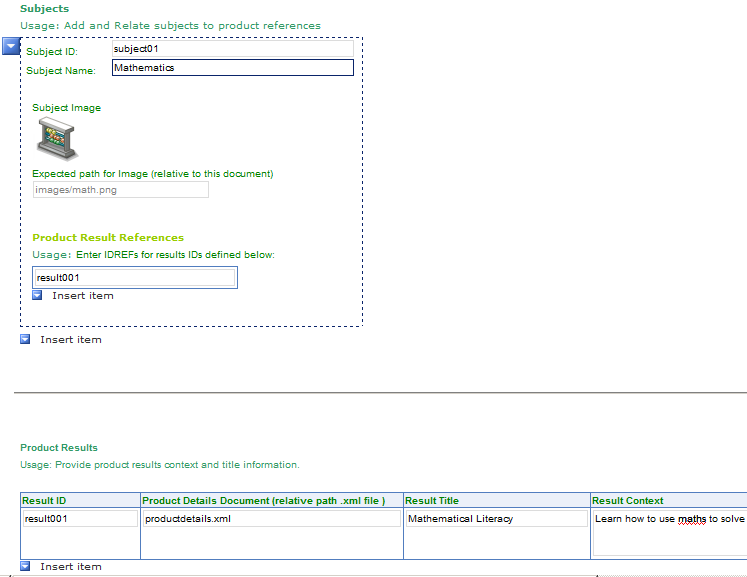
For any Telco provider support region, there should be 1 subject areas document or other format ( master catalog ), which feeds the My Courses feature. For content providers, a single area to product document should be in the package to the service provider.

### Provider Authoring

The template to use for providers is SubjectAreas.xsn. A single provider should hand-off a single subject areas document ***subjectareas.xml*** within the handoff package. At a minimum, the subjectarea.xml should contain information pertaining to product within the package.

A provide should only provide a single subjectarea.xml document. The contents of the document should only show how a single product results is associated with a single subject area. For provider authoring steps, see the Quick Guide.

#### Example Subject Area Authoring Form



The following is the model of **SubjectAreas**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] version** | Optional simple float value representing a version of the same document |
| **Subjects** | 0-1 group of 1-100 Subject elements |
| **ProductResults** | 0-1 group of 1-1000 Product elements |

The following is the model of **Subject**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] id** | Required unique ID for the subject. |
| **Name** | Required , 40 characters, used for display in the caption area upon control focus |
| **Image** | Required. 1 Image element with src attribute for the subject icon |
| **ProductResultReferences** | Required.(see next) |

The following is the model of **ProductResultReferences**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] subjectareassrc** | Optional URI for source of product results |
| **ProductResultRef** | 0-1000 ProductResultRef elements |

The following is the model of **ProductResultReference**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] internalidref** | Required interal idref for a product result defined within the same document |

The following is the model of **ProductResult**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] productdetailssrc** | Required URI of the product details source document. |
| **ResultTitle** | Required 40 char title, used for display title within the name area of list |
| **ResultContext** | Required 100 char string, used to marquee additional information about the result |
| **ProductResultReferences** | Required. Group |

## A Note about Inner-Topic Navigation (Activity to Activity navigation)

The content outline defines the navigation items for My Learning Course Overview, Lesson Outline, and Topic Outlines, however, inside a topic, the user will need to navigate further from screen to screen within a topic interact with all the topic content. We are calling this “inner-topic” navigation.

Authors define inner topic navigation as an attribute at the root level of the activity content types by providing IDREF values for the **nextidref,** and **previdref** attributes.

### Location Notification

On Text and Picture content types, authors can also specify a related Location for the page. The application will show or ( blink ) a notification indicator and enable a menu item to access the location feature of the application. Authors specify a single location IDREF at the root of the document. Content authors can include a **locationdetails** and **locationdetailsidref** at the top of a text and picture topic page.

## ContentAssets

There should be 1 contentassets.xml document per ***product***.

### Note about content for mobile devices

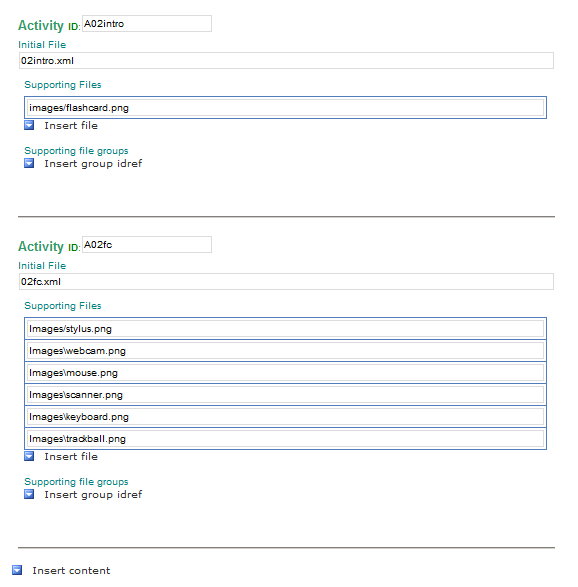
Mobile content handling is different than your typical desktop application or browser-based application experience. For mobile content, the content views, while more concise, still typically don’t contain anywhere near the amount of information presented in desktop views. This means the content data demand for each view is smaller and the content data transmission can also be smaller. This is good because mobile content needs to help support a model for delivery of experiences under seriously restricted bandwidth or network latency. As well, the mobile application needs to be aware of client data purge requirements for small devices, which help memory and space management on small devices. So, different from IMSManifest, OEB, or other manifest use, content assests and content organization defined here are smaller and enable more granular separation.

Authoring a content assets document with proper identification of these dependencies, enables compatibility to SCORM 2001 content compliance content asset management.

Note: will help with later backwards compatibility to SCORM-based systems.

A content assets document defines the xml and image file references to be used for individual content experiences.

Editing Template: ***ContentAssets.xsn***



The following is the model of **ContentAssets**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] version** | Optional simple float value representing a version of the same document |
| **FileGroup** | Can occur 0-999 times and is an arbitrary grouping of files, used to created common file sets as a reference item to be used within a ContentAsset\SupportingFiles section, using IncludeFiles element |
| **ContentItem** | Occurs 1-9801 ( max number available for an outline). These define content assets for a presentation. |

The following is the model of **FileGroup**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] id** | A unique identity within the document |
| **File** | Occurs 1-999 times, A *FileReference* with corresponding ***src*** attribute. |

The following is the model of **ContentItem**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] id** | A unique identity within the document |
| **InitialFile** | Required 1 time, A *FileReference* with corresponding ***src*** attribute. |
| **SupportingFiles** | Occurs 0-1 times, A group of ***File*** or ***IncludeFiles*** elements. |

The following is the model of **File**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] src** | A common URI form |

The following is the model of **IncludeFiles**:

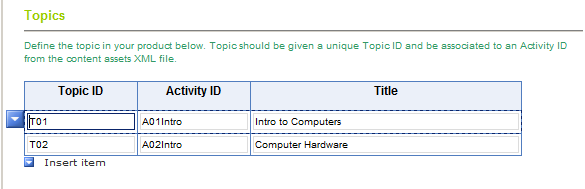
|  |  |
| --- | --- |
| Name | Description |
| **[attr] idref** | A reference to a unique identity within the document ( a FileGroup element ) |

## Content Outline

A content outline document defines the order of the content lessons and topics which would be discoverable for navigation.

There should be 1 content outline document per ***product***.

Editing template: ***ContentOutline.xsn.***



The following is the model of **ContentOutline**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] version** | Optional simple float value representing a version of the same document |
| **[attr] contentassets** | (defaults as “**contentassets.xml**”) URI for the file containing ContentAsset identities reference here ( expecting a partial file name here silbling, descendant, or sibling/descendant to the contentoutline.xml document |
| **Items** | Occurs 1 time and is a grouping element for items |

The following is the model of **Items:**

|  |  |
| --- | --- |
| Name | Description |
| **Item** | A navigable item within the content. Can occur 1-99 |

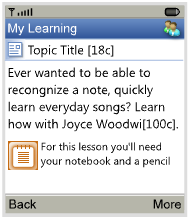
The following is the model of **Item:**

|  |  |
| --- | --- |
| Name | Description |
| **[attr] id** | ID type, represents a unique navigation ID within the content |
| **[attr] contentassetsidref** | String type, representing a unique content asset id (used for upload) |
| **Title** | An 40 character string used for the title of the content item. |

## Text and Picture Content

This is a reading content type document which provides a few different views of text and picture information:

##### “Introduction” Activity



This activity view is recommended for introducing another inner-topic activity to the learner. It contains:

Contains an icon with black short title ( 20c )

paragraph(s) (220c)

optional reader aid ( 30x30 PNG, and text 100c )

Editing template: ***TextAndPicture.xsn***.



##### “Summary” Activity

##### 

This activity is used for course overview as well as for single heading and paragraph content. Terms and definitions would also fit into this model. It contains:

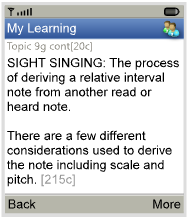
Contains a grey short title ( 20c )

A colored heading ( 20c )

paragraph(s) (220c)

Editing template: ***TextAndPicture.xsn***.

##### “Continued” Activity



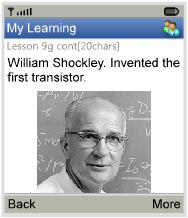
Provides a reading experience without the icon. This would be used for continuing text reading activities - to break apart long reading passages. It contains:

Contains a grey short title( 20c )

paragraph(s) (220c)

optional reader aid ( 30x30 PNG, and text 100c )

##### “Captioned Image” Activity



This would be used to provide a caption/titled image to the reader. Image size is contrained to 114 pixels here and best for photos or renderings. Use PanImage instead for thin line art illustrations. It contains:

Contains a grey short title ( 20c )

A caption (60c)

An image ( 114x114 PNG )

All **TextAndPictureContent** documents contain the following

|  |  |
| --- | --- |
| Name | Description |
| [attr] version | Optional version number for document ( reserved for tracking purposes ) |
| [attr]contentassetssrc | Optional reference to a contentassets document ( defaults to contentassets.xml ) |
| [attr]previdref | Optional string reference to an content asset within a contentassets document |
| [attr]nextidref | Optional string reference to an content asset within a contentassets document |
| (self) | Exactly one 1 of the following ***Summary, Introduction, Continuation, or CaptionedImage*** |

The following is the model of **Summary**:

|  |  |
| --- | --- |
| Name | Description |
| TopicTitle | Required 1-20 character representation of the topic title ( truncated or abbreviated as needed ) |
| Heading | 1 required 0-20 character subhead, rendered in accent color, small text. |
| Text | 1 required text element containing 1-220 characters of whitespace preserved text. |

Summary is intended to be used for both a course overview page content ( using the optional TitleAndByline ), and form simple **term and definition style content**.

The following is the model of **Introduction**:

|  |  |
| --- | --- |
| Name | Description |
| Icon | Require URI for a **20x20** pixel png format icon representing the topic type |
| TopicTitle | Required 1-20 character representation of the topic title ( truncated or abbreviated ) |
| Text | Required 1-220 character whitespace preserved text string |
| ReaderAid | Optional reader aid |

The introduction element is mainly used as a topic landing page, before the main activity, and can be used to quickly identify the type of activity with a small heading and graphic to the user. The remainder of the content on the page provides a terse model for reader preparation purposes using a small text field and a reader aid.

The following is the model of **ReaderAid**:

|  |  |
| --- | --- |
| Name | Description |
| Icon | Require URI for a **30x30** pixel png format icon representing the topic type |
| Text | Required 1-100 character string ( whitespace normalized ) text |

The intent of the reader aid is to provide a short notification to the user about upcoming content, such a preparation materials, or prerequisites.

The following is the model of **Continuation**:

|  |  |
| --- | --- |
| Name | Description |
| TopicTitle | Required 1-20 character representation of the topic title ( truncated or abbreviated as needed ) |
| Text | Required 1-220 character whitespace preserved text string |
| ReaderAid | Optional reader aid |

The intent of the continuation page is to provide a simple text reading experience, without inline formatting. An optional reader aid is allowed at the bottom of the text area.

The following is the model of **CaptionedImage**:

|  |  |
| --- | --- |
| Name | Description |
| TopicTitle | Required 1-20 character representation of the topic title ( truncated or abbreviated as needed ) |
| Caption | Required 1-50 character whitespace normalized text string |
| Image/[attr]src | Photo realistic or rendering 114x114 PNG format image |

## Flashcard Game Content

This is a content document which authors a flash card user experience.

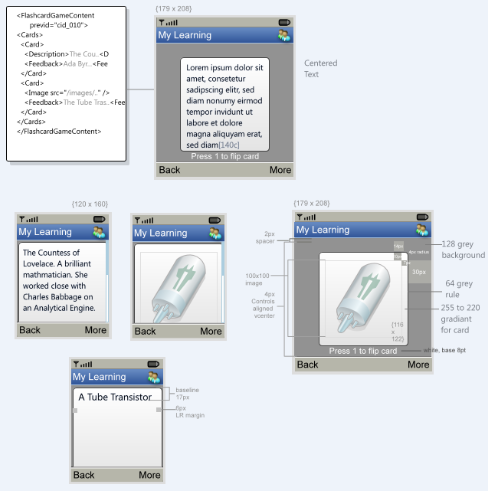
Provides a user experience for a short set of flashcards. The face of the card can contain an image or text description, the back of the card

A description ( 200c )

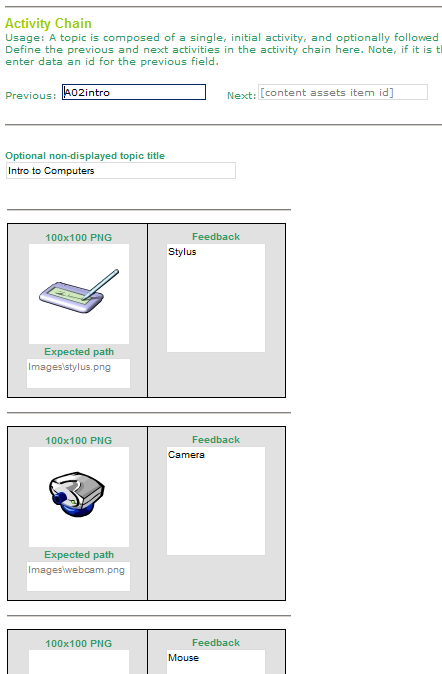
OR

An image ( 100x100 PNG )

                Feedback text (60c)



Editing template: ***FlashcardContent.xsn***.



The following is the model of **FlashcardGameContent**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] version** | Optional simple float value representing a version of the same document | |
| **[attr]contentassetssrc** | | Optional reference to a contentassets document ( defaults to contentassets.xml ) |
| **[attr]previdref** | | Optional string reference to an content asset within a contentassets document |
| **[attr]nextidref** | | Optional string reference to an content asset within a contentassets document |
| **TopicTitle** | | Optional ( non displayed ) 1-60 character title string |
| **Cards** | | 1 group of Card elements |

The following is the model of a **Card**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **(choice group)** | Required select 1 of Description or Image | |
| **Feedback** | | Req 1-60 character string for the text on the back of a card. |

The following is the model of **Description**:

|  |  |
| --- | --- |
| Name | Description |
| **(self)** | Req 1-200 character string for the front of the card |

The following is the model of **Image**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr]src** | Required URI for the local source of the image |
| **[attr]finalsrc** | (Form modified) relative path representing a folder/filename for the image |

## Adventure Game Content

This is a content document which authors a scenario walkthrough experience. Provides a text-based scenario exploration experience where a scenario and text choices leading to other scenarios are provided. It contains:

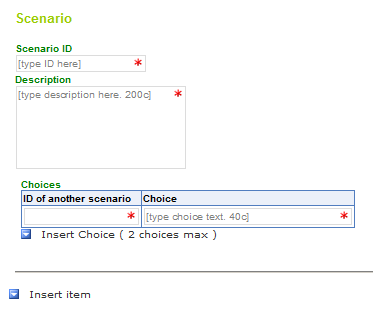
For each scenario:

A scenario description (200c)

1-3 choice phrases ( 40c each )

( no comp yet )

Editing template: ***AdventureContent.xsn***.



**DEV NOTE**: an adventure should be all within the same content file – no more than 15 description and choice pairings will be in the same content file ).

Adventure begins displaying the first scenario found with the content document. The uses selects using up/down, soft selection ( aka a normal drill down list ). The scenario displayed then changes based on the idref used for the selected choice item. Adventures end, when no choices are provided ( a terminal scenario ).

They navigate to the next activity using menu or key selection.

The following is the model of **AdventureGameContent**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] version** | Optional simple float value representing a version of the same document | |
| **[attr]contentassetssrc** | | Optional reference to a contentassets document ( defaults to contentassets.xml ) |
| **[attr]previdref** | | Optional string reference to an content asset within a contentassets document |
| **[attr]nextidref** | | Optional string reference to an content asset within a contentassets document |
| **Title** | | Req 1-22 character string. |
| **Scenarios** | | Required 1 group for 1-15 Scenario elements |

The following is the model of **Scenario**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] id** | Required. ID for the scenario. | |
| **Description** | | Req 1-20 character string. |
| **Choices** | | Optional 0-1 group for Choice elements |

The following is the model of **Choice**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] idref** | Required. IDREF type leading to another scenario. | |
| **(self)** | | Req 1-40 character string. |

## Sort Game Content

Provides a sorting experience providing a short list of terms/phrases and 2 categories to sort them by. It contains:

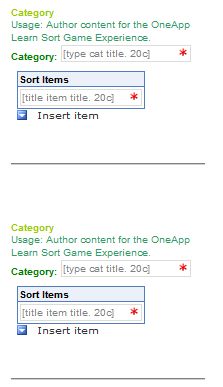
Two category phrases (30c).

1-10 terms (80c)



Editing template: ***SortGameContent.xsn***.

Editing Template: SortGame.xsn



The following is the model of **SortGameContent**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] version** | Optional simple float value representing a version of the same document | |
| **[attr]contentassetssrc** | | Optional reference to a contentassets document ( defaults to contentassets.xml ) |
| **[attr]previdref** | | Optional string reference to an content asset within a contentassets document |
| **[attr]nextidref** | | Optional string reference to an content asset within a contentassets document |
| **Categories** | | 1 required group of Category elements |

The following is the model of **Category**:

|  |  |
| --- | --- |
| Name | Description |
| **Title** | Required 1-URI for the local source of the image |
| **SortItems** | 1 req. group of 1-20 SortItem elements |

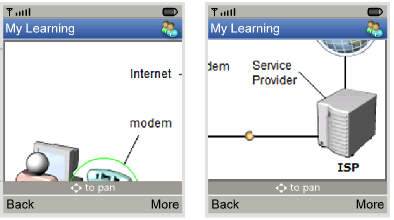
The following is the model of **SortItem**:

|  |  |
| --- | --- |
| Name | Description |
| **(self)** | 1-20 character string |

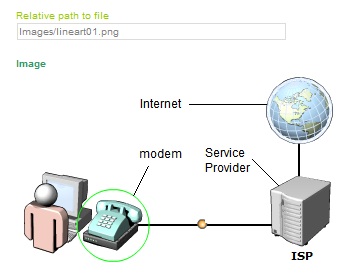
## Pan Image Content

Intended for larger or more detailed images such as line art or labeled screen shots, provides a 1:1 image presentation experience with panning functionality. This contains:

Image ( 128KB PNG file )



Editing template: ***PanImage.xsn***.



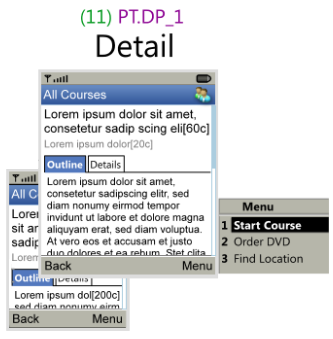
The following is the model of **PanImageContent**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] version** | Optional simple float value representing a version of the same document | |
| **[attr]contentassetssrc** | | Optional reference to a contentassets document ( defaults to contentassets.xml ) |
| **[attr]previdref** | | Optional string reference to an content asset within a contentassets document |
| **[attr]nextidref** | | Optional string reference to an content asset within a contentassets document |
| **TopicTitle** | | Optional ( non displayed ) 1-60 character title string |
| **Image** | | 1 required Image element |

The following is the model of **Image**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr]src** | Required URI for the local source of the image |
| **[attr]finalsrc** | (Form modified) relative path representing a folder/filename for the image |

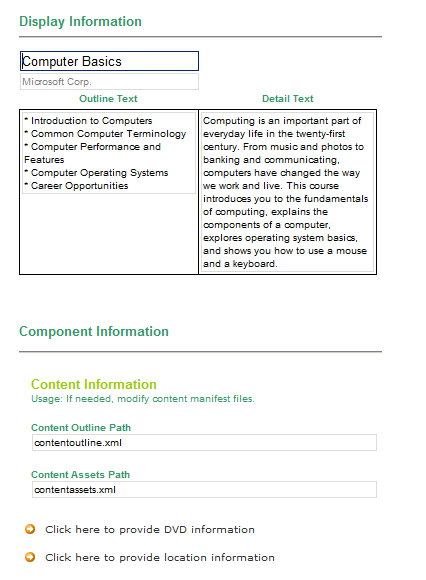
## Product Details



There should be 1 product detail document per product.

A product detail defines discoverable information about a product including: a Title, a byline, description, short outline.

Editing Template: ***ProductDetails.xsn***



The following is the model of **ProductDetails**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] version** | Optional simple float value representing a version of the same document |
| **[attr] lang** | String representation of a language code or name ( agreed upon standard ) |
| **Title** | Required 1-60 character product title |
| **Byline** | Required 1-20 character line for name of author, or provider |
| **Detail** | Optional 1-500 character, whitespace preserved text representing details about the product components |
| **Outline** | Optional 1-500 characters, whitespace preserved text representing the outline of the content |
| **Components** | Required 1 time. List of components |

The following is the model of **Components**:

|  |  |
| --- | --- |
| Name | Description |
| **OneAppContent** | 0-1 OneAppContent element |
| **DVD** | 0-1 DVD element |
| **Locations** | 0-1 Locations element |
| **AccessPrograms** | 0-1 AccessProgram element |

The following is the model of **OneAppContent**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr]contentoutlinesrc** | (Defaults to **contentoutline.xml** ) URI of product content outline file |
| **[attr]contentassetssrc** | (Defaults to **contentassets.xml**) URI of product content assets file |

The following is the model of **DVD**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr]sku** | Required string representation of a SKU number for ordering |

The following is the model of **Locations**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr]locationdetailssrc** | (Defaults to **loctiondetails.xml**) String representation of location details source content |
| **(self)** | A list of 1-100 Location elements |

The following is the model of **Location**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr]locationdetailidref** | Identity reference for a location element within a location details source |

The following is the model of **AccessPrograms**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr]offerdetailssrc** | (Defaults to **offerdetails.xml**) String representation of offer details source content |
| **(self)** | A list of 1-100 AccessProgram elements |

The following is the model of **AccessProgram**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr]offerdetailidref** | Identity reference for a program access element within a offer details source |

## Location Details

Location details defines location data about an iCafe, community center, school, etc.

The Location catalog has a few views which provide different levels of location information:

### Location Town View

This view is accessed from the Learning App spash screen menu. It contains a list of local/provider specific cities which are associated to facility locations.

### Location Results View

This view uses a list with an expandable focus item. Non-focued items would show the name of the location. Focused items would show, the name, and a “munged” text display of the important, required ( not-all ) location details. Activating the focus item opens the location detail view for the item

### Location Detail View

This view provides the entire set of location data for a location including name, address, optional description, telephone number, and optional website.

### Location Notification

On Text and Picture content types, authors can also specify a related Location for the page. The application will show or ( blink ) a notification indicator and enable a menu item to access the location feature of the application. Authors specify a single location IDREF at the root of the document. Content authors can include a **locationdetails** and **locationdetailsidref** at the top of a text and picture topic page.

### Forms and uses of LocationDetails.xml documents

**RUN-TIME:** The first form is intended for use by system administrators, and would be used to create the location town view. It implements the TownRegion content model elements as well as provides references to a single Locationdetails subdocument. *See, LocationDetailsSample1.xml.*  
An optional form, the second form would be to provide a subset of Location model elements representing the locations within a Town/Region… (optimized in terms of its contents as needed). *See, LocationDetailsSample2.xml.*

**AUTHORING:** The third form would be used by location providers to submit location information and host full location detail information. This could be provided in segmented form for performance requirements. *See, LocationDetailsSample3.xml.*

Editing template: ***[TBD]***.

The following is the model of **LocationDetails**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] version** | Optional simple float value representing a version of the same document | |
| **TownRegions** | | Optional 0-1 Group for TownRegions |
| **Locations** | | Optional 0-1 Group for Locations |

The following is the model of **TownRegions**:

|  |  |
| --- | --- |
| Name | Description |
| **TownRegion** | 1-99 TownRegion Elements |

The following is the model of **TownRegion**:

|  |  |
| --- | --- |
| Name | Description |
| **Name** | 1, 1-40 char Name of a Town or Region |
| **LocationReferences** | 1 req. LocationReferences element |

The following is the model of **LocationReferences**:

|  |  |  |
| --- | --- | --- |
| Name | Description | |
| **[attr]locationdetailssrc** | | Optional URI as the source of the locations to be used as a navigation result set. |
| **LocationRef** | 1-99 LocationRef elements | |

The following is the model of **LocationRef**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] inneridref** | Required string representation a Location id element within the same document |

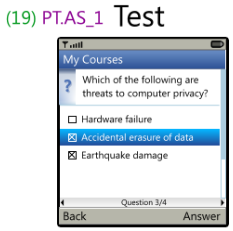
The following is the model of **Locations**:

|  |  |
| --- | --- |
| Name | Description |
| **Location** | 1-99 Location Elements |

The following is the model of **Location**:

|  |  |
| --- | --- |
| Name | Description |
| **[attr] id** | Required. A unique ID for the Location |
| **Title** | Required 1, 1-60 char string represents the name of the location |
| **Address** | Required 1, 1-60 char string for the address of the location |
| **Description** | Optional 0 to 1, 1-100 char string for the description of the location |
| **Telephone** | Optional 0 to 1, /d{3}-/d{3}-/d{4} pattern for phone number |
| **Website** | Optional 0 to 1, 60 character URI for website |

## Assessment Content ( Test )



An assessment document defines groups of assessment questions linked to topics and external content.

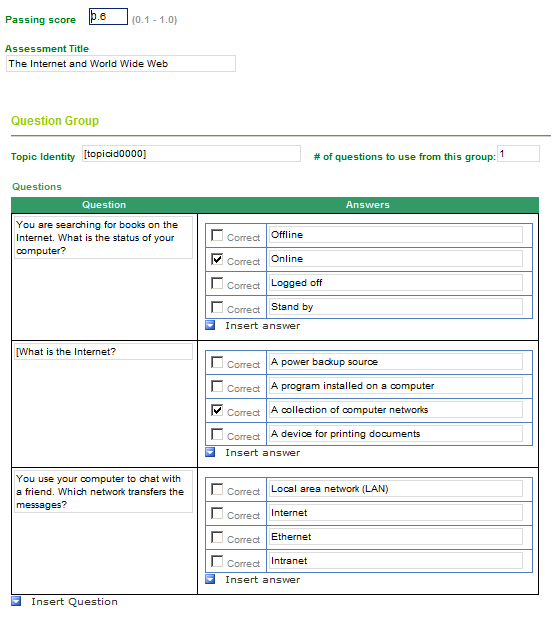
So, for each topic you want to test on, you define “questions pools” which are groups of questions and define a minimum number of questions to be used during the assessment from that pool.

Optionally, at the top of the assessment document, you define a passing score between 0.1 and 1.0. By default this will be .6.

There is only a single question type for assessments, ***multiplechoice, multiplecorrect***. So, for any given question *1 or more correct answers* may be given. For a student to answer the item correctly, they have to select *all the answers attributed as correct*.

If only one answer for a question is attributed as correct, then the behavior is similar to a *multiplechoice, singlecorrect* question. True and false questions can be written this way as well.

Editing template: ***Assessment.xsn***.



The following is the model of **AssessmentContent**:

|  |  |
| --- | --- |
| Name | Description |
| [attr] version | Optional version number for document ( reserved for tracking purposes ) |
| [attr] contentoutline | URI for the file containing ContentOutline topic identities |
| [attr] passingscore | Optional 0.1-1.0 floating point number for passing score ( default .6 ) |
| Title | Can occur 0-1, 1-60 character string |
| QuestionGroup | Can occur 1-100 times and is a grouping of content items |

The following is the model of **QuestionGroup:**

|  |  |
| --- | --- |
| Name | Description |
| [attr] topicidref | Identity of topic the question group relates to |
| [attr] questioncount | Number of questions to use from the group |
| Question | Can occur 1-100 times |

The following is the model of **Question:**

|  |  |
| --- | --- |
| Name | Description |
| Description | 1-60 character string used to pose the question |
| Choices | Occurs 1 time, contains 1-10 Answer items |

The following is the model of **Answer:**

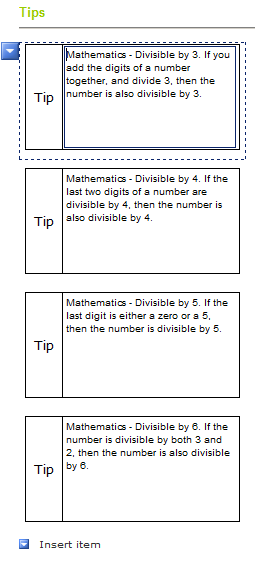
|  |  |
| --- | --- |
| Name | Description |
| [attr] correct | Default false, required as "true" to indicate a true answer |
| (self) | A 1-40 character string |

## Tip Content

The tip content is used to present a tip or word of the day on the home screen.



Editing template: ***Tips.xsn***.



Tips are randomly chosen from any product which contains tipcontent.xml within the package and where the product is also within the users My Courses list.

The following is the model of **TipContent**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] version** | (default 1.0). version for document | |
| **Tips** | | Required 1-50 group of Tip elements |

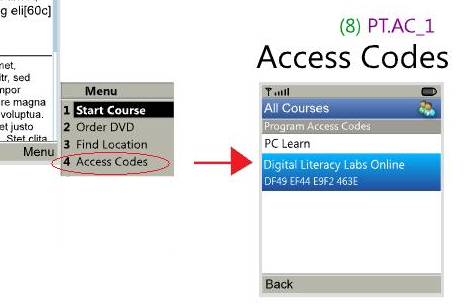
The following is the model of **Tip**:

|  |  |
| --- | --- |
| Name | Description |
| **(self)** | Req 1-220 character string. |

## Offer Details

**OfferDetails.xsd, OfferDetails.xml**:.

Program Access offers are available from detail pages which have an associated program access component.



Editing Template: ***OfferDetails.xsn***

The following is the model of **OfferDetails**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] version** | (default 1.0). version for document | |
| **ProgramAccessOffers** | | Optional 0-1 list of ProgramAccess items |

The following is the model of **ProgramAccess**:

|  |  |  |
| --- | --- | --- |
| Name | | Description |
| **[attr] id** | Require unique id for the program access offering. | |
| **ProgramName** | | Required 40 characters for the program name |
| **AccessCode** | | Require 60 characters for the program access code. |