SLK Core Concepts

Status: **Complete**
Last Updated: **12/8/2006 1:27 PM**

1. Summary

SharePoint Learning Kit (SLK) is a Learning Gateway 3.0 component that allows users to assign, execute, track and report on e-learning content inside SharePoint. It will run both SCORM and Class Server content, effectively bringing Class Server users into the Learning Gateway future. Unlike Class Server, SLK requires SharePoint (specifically v3). SLK has the following high-level features:

- Supports SCORM 1.2, SCORM 2004, Class Server LRM, Class Server IMS+ e-learning content formats
- Allows assignment of any non-e-learning content (e.g. Office documents)
- Supports multiple roles per user and learner self-assignment
- Better support for large districts and hosted deployments
- Lightweight config and zero provisioning (due to tight integration with SharePoint—of course you will have to configure and provision SharePoint.)

This specification describes the core concepts of SharePoint Learning Kit. It provides an overview of the product and describes key elements that are common across the entire product. Reading this specification will provide a solid background for understanding the other product specifications.

1.1. Terminology

Word or phrase	Meaning
Assignment	A record of one piece of content from a SLK-enabled document library assigned to one or more learners. A piece of content may be assigned more than once with each being a separate assignment. Each assignment will have one learner assignment for each assigned learner.
Assignment List	The detailed list of assignments displayed in the Assignment List Web Part. This is the resulting list of assignments and associated column data returned from an SLK query.
Assignment List Web Part (ALWP)	This is the only web part in SLK and is used to display a list of assignments for the current user on the SharePoint site. See spec: Assignment List Web Part.
Computed Points	The score derived from automatic and manual scoring within the content. For LRM, the computed score is the sum of all LRM question instructor scores for all activities in the assigned content.
Final Points	The final score set in the assignment grading page for a learner assignment. This score defaults to the computed score if there is one, but the instructor may edit the score as desired to set a *final* score for the learner assignment.
Group	A collection of users in SharePoint. SharePoint uses two types of groups, Domain Groups and SharePoint groups. Each SPSite has some number of Groups.
Instructor	A user who can assign to others, track, and grade an assignment in SLK.

Learner	A user who is attempting or may attempt an assignment in SLK. In the context of an SPWeb, a Learner is a user who may be assigned an assignment in SLK.
Learner Assignment	One learner's record of an assignment. When an assignment is created, a Learner Assignment is created for each assigned learner. Each learner assignment has a corresponding learner assignment state.
Learner Assignment State	The workflow state of one learner's attempt of an assignment.
SLK-Enabled Document Library	A SharePoint document library that has been enabled with SLK functionality and displays the "E-Learning Actions" option in the content context menu.
LRM Question Auto-score	A score computed automatically for an auto-graded question in LRM content.
LRM Question Instructor score	A score set by the instructor for questions within LRM. For auto-scored questions, the instructor score is initialized to the auto-score and the instructor may modify it if necessary.
Points	The normalized unit of measure displayed by SLK to represent the "score" of a learner assignment.
Points Possible	The number of points possible for an assignment. The actual score is allowed to exceed this number.
Right	A unique SharePoint action that a user or group may be allowed to perform. Users and groups are not assigned rights directly, but are assigned them in aggregate through the assignment of a "role definition". SharePoint has a fixed set of rights that are not extensible.
Role Assignment	The matching of one or more role definitions to a group or user. A group or user is not assigned individual rights, but is only assigned permissions through a role definition.
Role Definition (Permission Level)	A collection of rights that may be applied through role assignment to a group or user. A role definition is scoped to the SPWeb, but may be inherited from the parent SPWeb. For example, Contribute is one of the default role definitions in SharePoint.
Self-Assignment	The act of creating an assignment by directly launching the content from the e-learning actions page. An assignment is considered self-assigned if there is only one learner, zero instructors, and the learner is also the creator of the assignment. Any user who has access to the content in a SLK-enabled document library may create a self-assignment. It is not limited to users with the instructor role like a standard assignment is.
SharePoint Central Admin	SharePoint central administration contains links to administer SharePoint. The SLK configuration page is added to SharePoint Central Admin.
SLK instructor role	The role definition that SLK uses to identify users as instructors. It is recommended to use a role called "SLK Instructor" for this purpose; however, administrators can use any SharePoint role as the instructor role.
SLK learner role	The role definition that SLK uses to identify users as learners. It is recommended to use a role called "SLK Learner" for this purpose; however, administrators can use any SharePoint role as the learner role.
SLK Query	A query to the SLK database defined in the SLK settings file as a block of xml that specifies a base Learning Store view, set of conditions that must be met, set of columns to display, unique query name, title, and an optional sort.
SLK Query Set	A collection of SLK queries available in the Assignment List Web Part. This set defines the order of queries to be presented in the assignment query list as well as the default query displayed in the web part. Query sets are defined in the SLK settings file.

SLK Settings File	An xml based file that sets SLK configuration information. This file defines the SLK queries and query sets for the assignment list web part.
SPSite	The high level "root" collection of SPWebs in SharePoint. SPSites may not contain other SPSites, but may contain any number of SPWebs. A virtual server may have multiple SPSites. It is referred to as a site collection within the SharePoint UI. SLK specs use SPSite to avoid confusion with SPWebs.
SPWeb	A web site in SharePoint. A SPWeb may contain other SPWebs in a nested arrangement. It is referred to as a site or web site in the SharePoint UI.
Standard Assignment	An assignment using the instructor-learner workflow. The assignment is created by selecting "assign to others" on the e-learning actions page. This is different from a self-assignment in that the user may scope the assignment to a different SPWeb and can assign to other users. The user must have the instructor role on the target SPWeb in order to create a standard assignment.
Sub-web	An SPWeb that is a child to another SPWeb.
Summary List	This is the list of the query titles and their result counts displayed in the ALWP. Users may click on a title to run the query and display the results in the assignment list. The queries listed are the queries in the selected SLK Query Set.
User	An individual user who has been authenticated in SharePoint.

1.2. How to Read SLK Specs

1.2.1. Functional spec organization

We have the following functional specifications for SLK:

- SLK Core Concepts – Describes the foundation elements of SLK
- SLK Assignment Workflow UI –ASPX pages that provide the instructor and learner workflow
- Assignment List Web Part – The SLK web part used to view a list of assignments in SLK
- SLK E-Learning Frameset – Displays e-learning content in a web browser
- MLC SCORM Implementation – Clarifications on SLK's SCORM implementation
- Installation and Configuration – The SLK installation and configuration process and UI

1.2.2. M3 DCRs

Because this is a schedule-driven project, we have maintained a streamlined feature set for version 1 of SLK. We are developing in M1-3 the minimal feature set with which we can ship the product. The features specified in these milestones are the priority features.

Any additional non-critical feature or improvement is specified as a Milestone 3 Design Change Request ("M3 DCR"), and will be considered in M3 as time permits. M3 DCRs are called out throughout the functional specs, and a table of the DCRs is located at the end of each specification.

1.2.3. Design Decisions and FAQs

Each specification contains a list of commonly asked questions about the feature and descriptions of key design decisions. The intent is to provide transparency and background for design decisions. This should help with understanding the product and making future design decisions.

1.2.4. Specification History

Each specification contains a detailed history outlining how the spec has changed through time. The history provides insight into previous design considerations and how and why the design has changed.

2. SLK Installation and Configuration

SLK is designed for simple installation and configuration using standard SharePoint installation and configuration concepts. SLK is installed as a standard SharePoint solution and configuration is handled through SharePoint Central Administration.

The Installation and Configuration specification will describe this process in detail.

2.1. WSS Solution Deployment

SLK installs as a SharePoint Solution. WSS 3.0 provides new functionality by which SharePoint add-ins can be installed and deployed across an entire server farm from a single front-end server. SLK takes advantage of this new functionality to simplify the deployment process.

2.2. Configure SharePoint Learning Kit

After SharePoint Learning Kit is installed and deployed, administrators can configure SLK via the SharePoint Learning Kit SharePoint central administration page. SLK is configured per site-collection (SPSite). The single configuration page configures the following:

- Associates the SPSite to a SLK database, where assignment information will be stored
- Sets the SharePoint roles used to identify learners and instructors for SLK functionality
- Optionally sets modified SLK settings

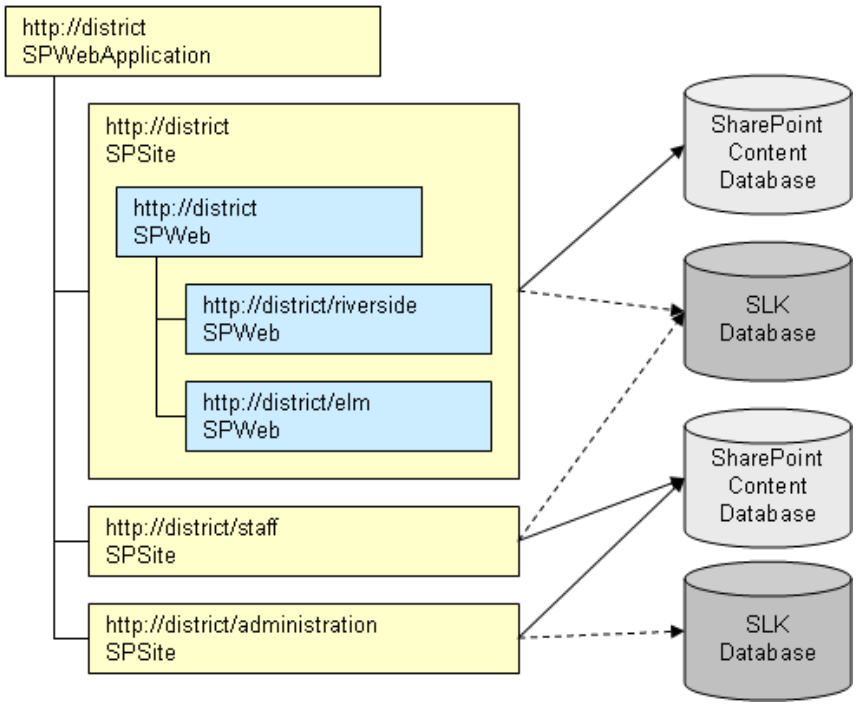
2.3. SLK Settings File

The SLK settings file is an xml file that can be uploaded in the SLK central administration page to set the SLK settings of an SPSite. There is a standard SLK settings file used by default for all installations, but administrators can apply various setting customizations using this file.

3. Provisioning and Authorization

SLK leverages the SharePoint site structure and common SharePoint site concepts. SLK does not have separate “class” provisioning or separate user authentication. Instructors and Learners are identified using standard SharePoint role assignments. If you are used to thinking about “class” site, in SLK you can consider any SPWeb a class site. SLK does not have a rigid template that must be used, but rather can be configured to work on any SPWeb.

3.1. Sample SharePoint Site Hierarchy



The image above illustrates the commonality between the SLK design and the SharePoint design. In SharePoint, every SPSite is associated with a SharePoint content database. The content database contains all data generated in all of the SPWebs on the SPSite. A content database can contain data for multiple SPSites, but each SPSite may store its data in only one content database.

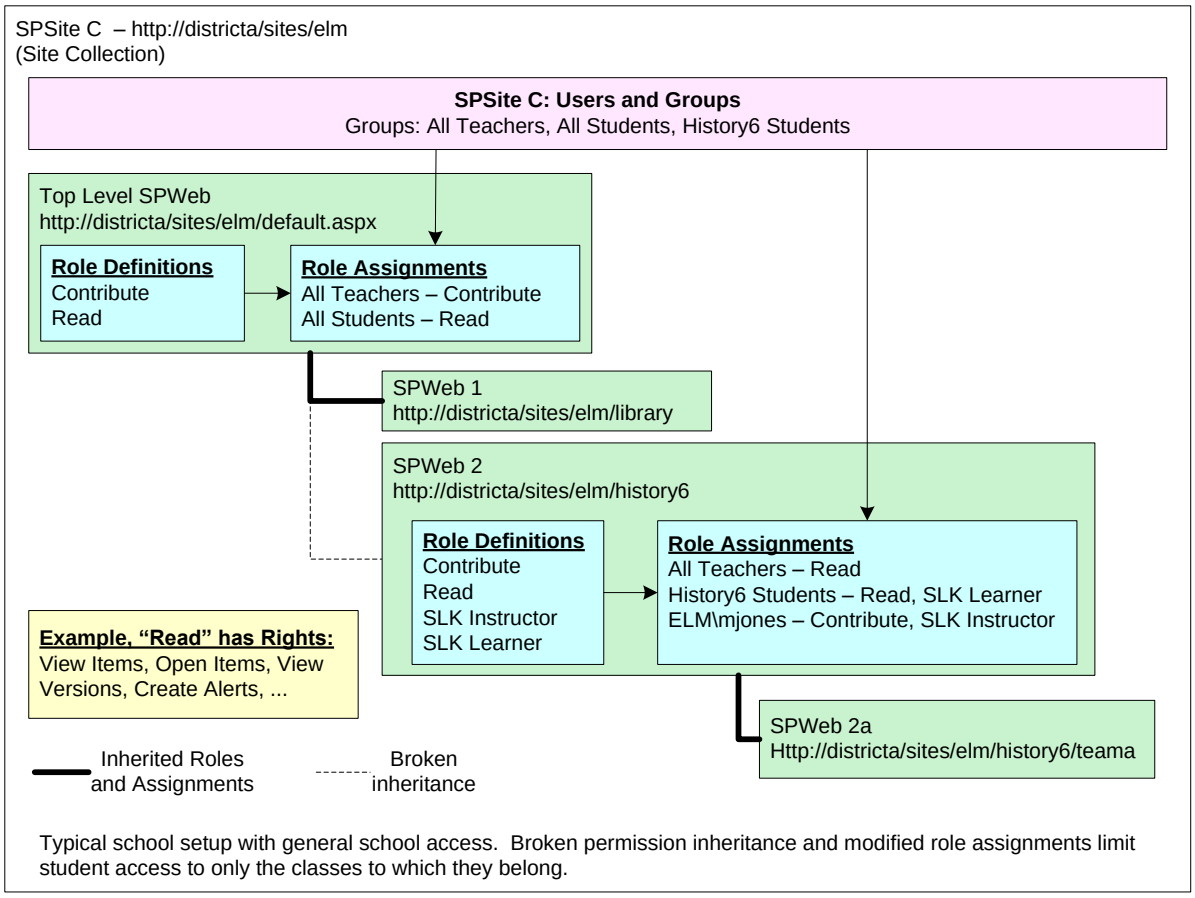
SLK has a database that stores assignment information. The actual assigned documents are stored in the SharePoint content database, but the SLK database stores assignment information, including assignment metadata, who the learners and instructors are for each assignment, and all of the data for each learner's attempt of each assignment.

The SLK database to SPSite relationship is similar to that of SharePoint's content database. Each SPSite stores its SLK data in a single SLK database, but one database can contain more than one SPSite's data.

In SLK, it is often advantageous to use the same SLK database for all related SPSites. SLK allows users to see assignments across the entire database, so if users on one SPSite are likely to access assignments scoped to another SPSite, both of those SPSites should be configured to use the same SLK database.

3.2. SharePoint Authorization

This section provides some background on how SharePoint handles authorization. SharePoint WSS V3 has the following authorization structure that is described in the subsections below.



3.2.1. Users and Groups

A given SPSite has a set of groups and users associated with it. The groups may be SharePoint groups or domain groups. SharePoint groups are managed in SharePoint and domain groups are managed in Active Directory. The set of groups and users are shared among all SPWebs of the given SPSite. Thus, a single SPWeb does not have a unique set of users and groups.

3.2.2. Rights and Roles

A SharePoint installation has a list of **rights** that are uniquely controllable actions within SharePoint. Users and groups are not assigned these rights directly. All authorization management occurs through role assignments, which assign a role definition to a user or group in a particular scope.

In SharePoint, a **role definition** is a list of rights associated with a role. For example, the Limited Access role in SharePoint has the following rights: View Application Pages, Browse User Information, Use Remote Interfaces, Use Client Integration Features, and Open. Role definitions are scoped to a particular SPWeb, but may be inherited from the parent SPWeb.

While rights are not extensible, new role definitions can be created by an administrator. New role definitions must be created at the root of the role inheritance. Thus, you may not create a new role definition scoped to a SPWeb that is inheriting its roles from a different SPWeb.

For example, if SPWeb1 is inheriting its roles, you may not inherit those "base" roles and also scope new roles to SPWeb1. To add the new role definition, you would either scope the role to the SPWeb at the root of the inheritance, or you could break the inheritance and scope the role to SPWeb1.

Role assignment matches one or more role definitions to a user or group. Role assignments are inherited from the parent SPWeb, but the inheritance may be broken to provide unique role assignments for any SPWeb.

Domain groups are treated in SharePoint much like individual users. Roles can be assigned to a domain group individually, or by adding the domain group to a SharePoint group that has the role assigned. For example, a site administrator can add a domain group Redmond\123abc to the Riverside Members SharePoint group to get the contribute role for the site. Unlike SharePoint groups, domain groups may be nested and contain additional domain groups.

Consider the following scenario. Assume you have the default site structure where all SPWebs are inheriting role assignments from the root SPWeb on the SPSite. You want SPGroup1 to have the Read role across all SPWebs in the SPSite except for SPWeb1 and all its sub-sites, where you want it to have the Contribute role. To do this you would need to break role assignment inheritance on SPWeb1 and then change the role assignment for SPGroup1 from Read to Contribute.

3.2.3. Role Definition and Role Assignment Inheritance

Role definitions and role assignments maintain different inheritance. You may break the role assignment inheritance, but maintain the role definition inheritance.

For example, an administrator can break the role assignment inheritance to have different role assignments for groups and users on a SPWeb, but may maintain the role definition inheritance so the same roles are available for assignment on all SPWebs in the SPSite. Separately, if the administrator needs different role definitions on a SPWeb, she could break the inheritance of the role definitions and create the new role on the SPWeb.

This is useful information because if the administrator chooses to create instructor and learner role definitions (permission levels) during the configuration process, SLK will create those roles at the SPSite level and will rely on role definition inheritance to make those available to all the SPWebs of the SPSite. Even if role assignments are broken at the SPWeb level, which will be a common scenario for SLK users, the created role definitions will be available for the new role assignments on the SPWeb.

Note if any SPWebs have broken role *definition* inheritance prior to the SLK configuration process, those SPWebs will not automatically have the new instructor and learner role definitions.

3.3. SLK Authorization

In SharePoint Learning Kit there are several features for which the underlying components must get the following information efficiently in the scope of a SPWeb:

1. Which users are "Instructors" (either individually or as a member of a group)?
2. Which users are "Learners" (either individually or as a member of a group)?
3. Is the current user an "Instructor"?
4. Is the current user a "Learner"?

Items 1 and 2 happen in the assignment create/edit page (see SLK Assignment Workflow UI). Items 3 and 4 happen on pages containing the Assignment List Web Part (ALWP), at least once per ALWP per page view in order to display the appropriate query set for the user's role.

SLK uses SharePoint role definitions to identify instructors and learners in the scope of a SPWeb.

To determine the instructors in the scope of a given SPWeb we enumerate all groups assigned the instructor role and union all individual users assigned the role.

Similarly, to determine the learners in the scope of a given SPWeb we enumerate all groups assigned the learner role and union all individual users assigned the role.

SLK can also query SharePoint to determine if a given user is assigned a role in the scope of a SPWeb. For instance does Bob have a particular role for SPWeb1.

3.3.1. Specifying Roles

During the configuration process, an administrator may specify which SharePoint roles SLK should use to identify instructors and learners on the SPWebs of the configured SPSite. An administrator may set one role for instructors and one role for learners. They can use the same role for both learners and instructors if desired.

Rather than using existing SharePoint role definitions, the administrator may also specify new roles and have SLK automatically create those roles on the SPSite.

We recommend that new installations create custom roles called "SLK Instructor" and "SLK Learner" and assign those roles to users and groups to identify them as instructors and learners for SLK functionality. This is the default behavior during the configuration process.

3.3.2. Adding SLK Role Definitions

If the administrator elects to create new role definitions, the configuration process adds the roles at the root SPSite level and does not create them for any SPWebs that have broken role *definition* inheritance. The configuration process is covered in the Configuration spec.

When SLK roles are created, they are created according to table 1 below. The "Description" column contains the actual text to appear in the UI for these roles. The name is specified by the administrator on the configuration page.

Table 1: SLK Role Definitions

	Name	Description
Instructor	<Name specified in config form>	Can use SharePoint Learning Kit features to assign e-learning content to learners.
Learner	<Name specified in config form>	Appears as a potential learner when new assignments are created using SharePoint Learning Kit.

SLK uses these role definitions to identify users and groups with these roles, and they are created with no SharePoint rights. Site administrators may add rights to these roles without affecting SLK. It is expected that administrators will use other role definitions and role assignments outside the scope of SLK to apply SharePoint rights.

3.3.3. SharePoint Roles vs SLK Assignment Relationships

SLK uses the instructor and learner role definitions for the following purposes:

- Identify *possible* learners and instructors when creating or modifying an assignment
- Determine if the current user is an instructor and can create an assignment on a SPWeb
- Identify if the current user is a learner or instructor to determine which default query set to display in the assignment list web part

It is important to note that SharePoint role assignments are only used to determine availability of SLK features to certain users on an SPWeb. The role assignments do not affect SLK assignments. After an assignment is created, the learners and instructors for that assignment are maintained in the SLK database. The query results listed in the assignment list web part and the ability to access a given assignment or learner assignment are determined by the user's relationship to the assignment in SLK and are not dependent on their role in SharePoint.

For example, when a user tries to assign a document to others, SLK checks the role assignment on the target SPWeb to ensure the creator is an instructor. Then, on the create assignment page, SLK checks the roles on the SPWeb because all of the users with the learner role are listed as *potential* learners and all of the users with the instructor role are listed as *potential* instructors. The creator selects from these lists the learners and instructors for the assignment.

Once the assignment is created, SLK maintains the list of learners and instructors based on who was selected during the assignment creation process. If a learner is later removed from the SharePoint learner role for the SPWeb, their learner assignment still exists in SLK and can be accessed and seen in the assignment list web part. To remove a learner from an assignment, an instructor must modify the assignment and remove the user. That action deletes the user's learner assignment and associated work.

3.3.4. Authorization During Assignment Creation

To self-assign, the user must have access to the content being assigned, but is not required to have the instructor or learner permissions on the SPWeb.

To assign to others, the assignment creator must have access to the content being assigned and must have the instructor role on the SPWeb of the assignment. SLK verifies the instructor role for the creator at assignment creation time.

When verifying the current user has instructor permissions on a given site, the current user may have the permission directly, as a member of a SharePoint group with the permission, or as a member of a domain group with the permission. Note domain groups themselves may have the permission directly or as a member of a SharePoint group.

To be assigned an assignment, the learners and instructors are not required to have the learner and instructor permissions, but the permissions are used to list potential learners and instructors in the assignment creation UI.

For example, when the SLK API is used to create assignments for a group of learners, SLK will verify the assignment creator is an instructor, but will not check the roles of the assignment learners and instructors.

3.3.5. Assigning to Individuals, Selecting by Groups

When an assignment is created, SLK checks for all users and groups assigned the SLK learner role and lists them in the create/modify page. The assignment creator is then able to select users individually or by group to be learners for the assignment.

It is important to note that SLK only assigns to individual users and does not maintain groups of users in the schema. SLK uses groups in SharePoint to select individual users in the create/modify page. The key point here is that selecting a group on this page does not actually assign to that group, but rather assigns to all of the users in that group at that point in time.

For example, if a user is added to a SharePoint group, the user will not be assigned or unassigned as a learner to any assignments. To assign to the new learner, an instructor would need to modify the assignment and select the new user, who will now display in the available learners because of her new role assignment.

4. Queries and Views

4.1. SLK Queries and Query Sets

SLK uses Queries and Query Sets to display filtered lists of the current user's assignments in the Assignment List Web Part. This section describes queries and query sets. The Assignment List Web Part specification describes in more detail how they are used in the web part.

4.1.1. Goals

We have the following goals for specifying how SLK queries are defined and exposed to the web part administrator and the end user.

1. Provide a flexible method to display filtered lists of user assignments in the ALWP.
2. Allow administrators to modify and define new filtered database queries and display those results in the ALWP.
3. Have a web part property user interface for the web part administrator that allows them to define which query or query set is available to users, but does not require them to know specifically how queries and query sets are defined.
4. Allow query and query set definitions to be managed centrally, but allow individual web parts to set which query or query set it uses.

4.1.2. Definitions

Queries and query sets are defined as follows:

SLK Query – A database query that returns a list of assignments and specific data columns matching the query conditions. Queries are defined in a block of xml in the SLK Settings file that specifies:

- Unique name (across all queries and query sets) where spaces are not allowed
- Query title that is human readable and localized
- Base Learning Store View
- Set of conditions that must be met for the assignment to be displayed by the query
- Set of columns for the query to display in the assignment list
- Formatting attributes for the results displayed in the ALWP. (optional)
- Column sort (optional)

A SLK query starts with a base learning store view. It then filters the learning store view based on query conditions specified in the query definition resulting in the final set of records to be displayed in the assignment list. Finally, the query defines what data will be displayed in columns for each record in the list and also specifies formatting for the column data.

For example, the LearnerOverdue query may display all learner assignments that are overdue for the current user and display the Site, Assignment Title, Due Date, Status, and Score in the assignment list columns. Each column could have different text formatting.

SLK Query Set – A collection of SLK queries defined in a block of xml in the SLK Settings file that specifies:

- Unique name (across all queries and query sets) where spaces are not allowed.
- Query set title that is human readable and localized
- Set of SLK queries in order of presentation in the ALWP Summary List
- A default SLK query for the query set that will be displayed when the ALWP first loads with the query set.

The ALWP uses a query set to specify which queries should be displayed in the summary list and out of those, which query should be the default displayed in the assignment list when the ALWP first loads.

See the ALWP spec for more specific information on how queries and query sets are used in the web part.

4.2. LearningStore Views vs SLK Queries

LearningStore views are virtual tables that show a subset of data in the SLK database. They are defined in the schema as SQL SELECT statements that can return data from multiple items, return calculated values, and provide row-level security. The LearningStore view handles user authorization and only returns data that the current user can access as defined by the view. Each view is defined in the MLC database schema.

For example, the LearnerAssignmentListForLearners will only return assignments for which the current user is a learner on the assignment. Additionally, some data is processed prior to it being returned by the view. Final Points, for instance, is only shown to the learner after the assignment has been returned by the instructor. In this case, the LearnerAssignmentListForLearners definition specifies that in cases where the learner assignment state is less than “final”, return null for the final points value.

SLK Queries are defined in the SLK settings xml file and provide a flexible method for presenting data from a LearningStore view. A SLK query takes a LearningStore view and further filters it to display only the columns required in the format required for the query.

In general, LearningStore views return the proper result set from the database, and SLK queries format the data for presentation. The SLK SDK documentation has additional information about LearningStore views and SLK queries and query sets.

5. Content

5.1. Supported Content Formats

5.1.1. SCORM 2004

The SLK e-learning runtime engine supports SCORM 2004 content with full sequencing and navigation. SLK implements SCORM 2004 2nd Edition, Version 1.3.1 with the SCORM 2004 2nd Edition Addendum Version 1.2.

5.1.2. SCORM 1.2

SLK also supports SCORM 1.2 e-learning content. The e-learning runtime implements SCORM Version 1.2 with the SCORM 1.2 Addendum Version 2.0.

5.1.3. Class Server (LRM/IMS+)

SLK supports Class Server e-learning resources with the exception of the remote content and licensed content features listed below.

5.1.3.1. Remote content

The runtime for executing Class Server e-learning resources does not support the use of remote content. Class Server versions 3 and 4 supported a remote content protocol that allowed content publishers to host proprietary content separate from the e-learning package. This was not a highly used feature of Class Server, so we removed the added complexity from SLK.

5.1.3.2. Licensed LRM Content

SLK e-learning runtime does not support licensed Class Server resources. This was a feature included in Class Server that allowed content publishers to embed a license in the content that was required to execute the content in Class Server. SLK does not implement this feature and all content with embedded licenses will not work in SLK.

Those with licensed content can work with the content publisher on an agreement and steps to remove the license to execute the resources in SLK.

5.1.4. Non E-Learning Content

SLK supports assigning non-e-learning content. Generally, any document type that can be stored in a document library can be assigned in SLK. The SLKSettings.xml file contains the listing of MIME type mappings used in SLK and new MIME types can be added to support execution in SLK.

Non-e-learning content is managed through the same assignment workflow as e-learning content, but there is no learner interaction data stored in the SLK database as there is with interactive e-learning content. When a learner "Begins Assignment" with a non-e-learning resource, that resource is delivered via the appropriate client application or plugin as if they had clicked on a link in a web page.

To complete work on a non-e-learning resource and return it to an instructor for review, the learner will need to save a copy of the document locally and submit his/her work as prescribed by the instructor. There is no auto-save or drop-box feature in V1 of SLK. The learner can "Submit"

the assignment via the lobby page to indicate to the instructor that they are finished with the assignment.

5.2. Caching

When an assigned e-learning package is accessed for the first time, the standard SLK setting is to cache the package on the front-end server file system for faster access. Depending on the load balancing experience, it may be common for the first several accesses of the package after it is assigned to be slower than usual as the package is cached for the first time on each front-end server.

The file cache location and minimum expiration time are defined for each site collection and can be set in the SLK settings file. See the SLK Settings documentation for specifics on these settings.

5.2.1. Cache Location

In the standard SLK settings file, SLK does not specify a cache location. When a location is not set, SLK uses a package cache in the temporary directory on the front end server. In a standard installation, the same SLK settings file is used for all SPSites, so all SPSites share one cache in the temp directory of each front-end server.

Rather than using the default cache location, an administrator may specify a cache location in the SLK settings file. This is useful when using a file share location for a single cache shared by all front-end servers. Any SPSites configured with the SLK settings file will use the specified path. Note, since the cache location can be specified in the SLK settings file, it is possible for each SPSite to have its own cache for e-learning packages.

5.2.2. Cache Minimum Expiration Time

The minimum life of cached packages is also set for each site collection in the SLK Settings file. This is the minimum amount of time, in minutes, that a package will be kept in the cache. Packages not accessed in the cache within this amount of time are removed from the cache.

The standard SLK Settings file defines a minimum expiration time of 4320 minutes (3 days).

5.2.3. E-learning Package Security

SLK allows users to access an assigned document or package if they are a learner or instructor on the assignment. Managing access based on assignment relationship allows instructors to restrict access to the document library containing the curriculum packages, but still allow learners to access assignment content through the SLK assignment workflow.

5.2.3.1. Assigning a Particular File Version

When an assignment is created in SLK, a particular version of the package in a document library is used. Whenever the assigned content is accessed, if the assigned content is different from the current content in the document library, then SLK will display a general assignment resource not found error in the frameset (see SLK Frameset spec for details). This means that if versioning is turned off, editing a file will invalidate any assignments containing that file. Also, if versioning is on, but you are editing the assigned version, the same will be true. You can create new versions of

an assigned package and any assignments referencing the original version will continue to reference that version. Any new assignments will reference the new version.

5.2.3.2. Users Can Access Entire Package

It is important to note when assigning content that access is granted to the entire package rather than the specific organization assigned within the package. For example, consider a package that has two organizations and the second organization has resource files that the first organization doesn't use. It could be possible for the learner, through URL probing, to access the second organization's files even though the learner was assigned the first organization.

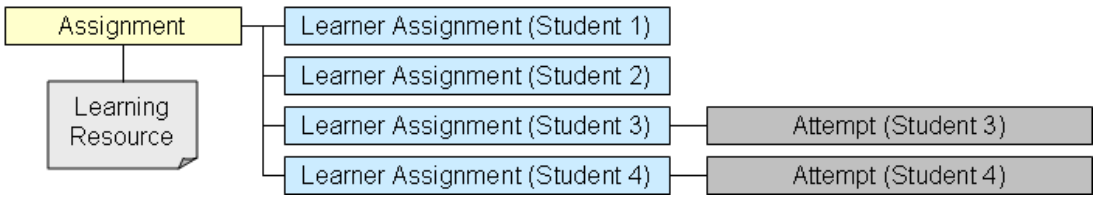
Instructors and content producers should be aware of this granularity of access restriction and develop and use e-learning content accordingly. SLK considers all files in an assigned package to be accessible to an assigned learner even though the assignment workflow UI will only present the assigned organization for delivery.

6. Assignment Workflow

6.1. Assignments and Learner Assignments

A single assignment can be created and assigned to multiple learners. When the assignment is created, a learner assignment is created for each of the learners. For e-learning assignments, each learner assignment references a specific attempt of the assigned activity. The attempt on the assigned package is not created until the learner starts their assignment. Non-e-learning content does not use attempt items.

In the figure below, students 1 and 2 have not started their assignments yet, but students 3 and 4 have.



6.1.1. Learner Assignment States

Each learner assignment has a workflow state called the *Learner Assignment State.* The SLK functionality available to learners and instructors is affected by the Learner Assignment State.

6.1.1.1. State Names and Descriptions

L.A. State	Displayed in UI ("Status")	Description
NotStarted	"Not Started"	The assignment has been created, but the learner has not "started" the learner assignment.
Active	"In Progress"	The learner has started their learner assignment, but has not yet "completed" their learner assignment. For e-learning

		content, the attempt status is "Active" or "Suspended".
Completed	"Submitted"	The learner assignment has been completed, either by the learner or collected by an instructor. • If the content is not auto-graded and auto-returned, then the learner assignment remains in this state until an instructor grades and returns the learner assignment. • In the auto-graded and auto-returned case, the learner will not see this state in the UI because the workflow will automatically skip over it. • For e-learning content, the attempt status is "Completed" or "Abandoned".
Final	"Final"	The learner assignment has been auto-graded and auto-returned, or manually graded and returned by an instructor. It is in its "final" state in the work flow.

6.1.1.2. State Naming Decisions

Learner assignment state names and their respective UI terms emphasize clarity to avoid confusion between the use of terms and their corresponding workflow stages. The internal state terms and the names displayed in the UI are different because we want the UI terms to closely match the learner workflow experience and we want the learner assignment state terms to remain distinct and map easily to attempt status. This allows for easier internal discussion of learner assignment state and their associated attempt statuses without confusing similar terms used for different purposes.

In particular there are subtle differences in the terms "completed" and "submitted". We internally use the term "completed" because it matches the attempt status and non-e-learning status when the content is "completed" or "abandoned". For the status displayed to the user we use the term "submitted" because it represents the workflow stage when an assignment has been completed, but is waiting for an instructor to grade and return the assignment. If a learner assignment is automatically graded and returned, then the user will not see the "submitted" status in the UI, it will immediately move to "final".

We use the term "final" rather than "returned" because not all assigned content has the concept of being graded and *returned*. Most SCORM content will simply be *done* and in a final state.

6.1.1.3. Determining the Learner Assignment State

The Learner Assignment State is determined differently for e-learning and non-e-learning content as described in the following pseudo code. The matrix that follows summarizes the inputs used to determine the learner assignment state.

```
if(RootActivityID is NULL)  //Non-e-learning content does not have an ActivityPackageItem
{
        //non-e-learning content
        If (NonELearningStatus is NULL)
                State = "Not Started"
        Else
        If (LearnerAssignmentItem.Final)
                State = "Final"
        Else
        If (NonELearningStatus == Active or Suspended)
                State = "Active"
```

```
        Else
                State = "Completed" //The NonELearningStatus is Completed or Abandoned
}
Else
{
        //E-Learning Content
        If (There is no AttemptItem)  //An Attempt has not yet been created.
                State = "Not Started"
        Else
        If (LearnerAssignmentItem.Final)
                State = "Final"
        Else
        If (AttemptStatus == Active or Suspended)
                State = "Active" //Attempt is Active or Suspended
        Else
                State = "Completed" //Attempt is Completed or Abandoned
}
```

Inputs to Determine Learner Assignment State					Learner Assignment State
Non-E-Learning Content		E-Learning Content		Both	
NonELearning Status Exists	**NonELearning Status**	**Attempt Exists**	**Attempt Status Property**	**LearnerAssignment Final Property**	
False	N/A	False	N/A	False	**NotStarted**
True	Active	True	Active	False	**Active**
True	Suspended	True	Suspended	False	**Active**
True	Completed	True	Completed	False	**Completed**
True	Abandoned	True	Abandoned	False	**Completed**
True	Completed	True	Completed	True	**Final**
True	Abandoned	True	Abandoned	True	**Final**

6.1.1.4. State Change Behavior

There are several areas in the SLK UI where the workflow state is advanced by learners, instructors, or content. The following table describes the internal state changes from workflow related actions.

Action	Behavior	
Begin assignment (learner)	E-learning	• Create and start the e-learning attempt. (AttemptStatus = Active) • Set LearnerAssignmentState = Active
	Non-Elearning	• Set NonELearningStatus = Active • Set LearnerAssignmentState = Active
Content – ExitAll	E-learning	• ExitAll the e-learning attempt. (AttemptStatus = Completed) • Auto-return behavior*
Content – AbandonAll	E-learning	• AbandonAll the e-learning attempt. (AttemptStatus = Abandoned) • Auto-return behavior*
Submit assignment (learner) Collect assignment	E-learning	• ExitAll the e-learning attempt. (AttemptStatus = Completed) • Auto-return behavior*
	Non-	• Set non-e-learning status = Completed.

(instructor)	Elearning	• Auto-return behavior*
Return assignment (instructor)	E-learning	• Set LearnerAssignmentFinal = True. • LearnerAssignmentState = Final
	Non-Elearning	• Set LearnerAssignmentFinal = True. • LearnerAssignmentState = Final
Reactivate assignment (instructor)	E-learning	• SCORM 2004 - Create a new attempt (AttemptStatus = Active) • Other content – Use same attempt, reset status to Active. • LearnerAssignmentFinal = False • LearnerAssignmentState = Active
	Non-Elearning	• NonELearningStatus = Active • LearnerAssignmentFinal = False • LearnerAssignmentState = Active

* The following auto-return behavior is triggered for each exit/abandon action:

```
If AutoReturn is true, then:
    LearnerAssignmentFinal = True
    LearnerAssignmentState = Final
else
   LearnerAssignmentState = Completed
```

The "AutoReturn…" property can be set at any time in the workflow. When this property is set, any "completed" LearnerAssignments are advanced to "final" the next time they are accessed.

DCR: Logging Learner Assignment State Changes

Whenever the learner assignment state changes, log the state change in a learner assignment event log. The learner assignment event log will be located in the SLK database and will be a record of all learner assignment state changes and access events. See Workflow UI spec for details on access events.

For state changes, the learner assignment event log will store the following fields:
- LearnerAssignmentItemIdentifier
- The current signed-in user
- State change action: Create, Begin, Submit, Collect, Return, Reactivate, or Delete depending on the state and the user's relationship to the assignment (learner or instructor)
- Old Learner Assignment State
- New Learner Assignment State
- Current date/time

6.1.1.5. Sample workflow and associated states

Action	Learner Assignment State after action	Displayed in UI
Instructor creates an assignment and assigns an LRM package. Student's Learner Assignment is created in this process.	NotStarted	"Not Started"
Student starts her learner assignment by launching the LRM package. (Her attempt is created at that time.)	Active	"In Progress"

Student saves her attempt and plans to return later to finish.	Active	"In Progress"
Student resumes her assignment and then submits it.	Completed	"Submitted"
The instructor grades assignment and returns it.	Final	"Final"
Student reviews her assignment and grade.	Final	"Final"

6.1.2. Learner Assignment States and View Access

SLK and MLC present e-learning content by session view. An e-learning session view determines how the e-learning content is presented to the user and what actions are available on that content. Each view has different runtime behaviors. SLK views correspond with their base MLC views as follows:

SLK View	MLC View	MLC View Description
Preview	Preview	Linear & choice sequencing, read only
Execute	Execute	Obeys sequencing rules, read/write
Grading	Random Access	Linear & choice sequencing, read/write
Learner Review	Review	Linear & choice sequencing, read only
Instructor Review	Review	Linear & choice sequencing, read only

In SLK, access to certain views of e-learning content are affected by the learner assignment state. The table below outlines the SLK views of a learner assignment available for each role and learner assignment state. See the SLK Frameset spec for specific access conditions for each view.

Learner Assignment State	**SLK views available in SLK UI**	
	Instructor	**Learner**
NotStarted	None	Execute (will start attempt)
Active	InstructorReview	Execute
Completed	InstructorReview and Grading	None
Final	InstructorReview and Grading	Review

6.1.2.1. If Instructor is also a Learner

For cases where the instructor is also a learner on the assignment, both sets of views can be available for each learner assignment state. The SLK Assignment Workflow UI specifies what views are available in the UI.

6.1.2.2. MLC Preview View

The MLC "Preview" view is used by SLK outside the scope of an assignment or learner assignment. MLC Preview view is available at any time without regard to learner assignment state.

Preview view will be accessible in the SLK UI from the Actions page.

6.1.2.3. Execute View

Execute view is the primary view when attempting e-learning content. A learner is not able to access the execute view in either the "Completed" or "Final" states.

6.1.2.4. Learner Review View

The learner uses this view to review their assignment after it is returned in the "final" state. A learner is not allowed to review an assignment prior to the "final" state because they could see correct answers or grading information.

6.1.2.5. Instructor Review View

The InstructorReview View is available any time the AttemptItem is not NULL. It is the view used when the instructor clicks on a learner's name in the grading chart prior to the learner assignment being submitted (LAState = Active).

6.1.2.6. Grading View

The view used when the instructor clicks on a learner's name in the assignment properties grading chart after the learner assignment has been submitted (LAState = Completed or Final). We use the grading view in the "final" state to allow instructors to modify grading after an assignment has been returned without having to reactivate the assignment to get back to grading view.

6.1.3. Reactivating a Learner Assignment

Learner assignments in the "final" state may be reactivated by an assignment instructor. Reactivating a learner assignment returns the assignment to the learner so they can modify their work and complete the assignment again. In general the learner's previous work is maintained, so they can modify the assignment from their previous state. Any grading changes made by the instructor are reset during reactivation so grading is consistent with the autograded values that correspond to the learner's work in the "In Progress" state.

For all reactivated learner assignments:
- Keep instructor comments for the learner assignment.
- Clear final points for the learner assignment.
- Update the learner assignment state and associated attempt status values to be "active" (In Progress).

Future workflow is unaffected by reactivation and proceeds from the active state. When the reactivated assignment is submitted/collected, we initialize final points according to the regular workflow process.

6.1.3.1. For E-Learning Content

When e-learning content (SCORM 2004, SCORM 1.2, Class Server) is reactivated, we maintain the previous learner attempt and attempt data, including any learner responses.

LRM
- LRM activities reset instructor evaluation scores to their current autograded value for the learner's response. For non-autograded questions, the evaluation score is cleared.
    - Note: This behavior is to "reset" the scores by clearing any instructor modifications. The updated score displayed in the graded score column (Total Points) will reflect the autograde prior to submitting the assignment. Any changes to the learner's responses in the

autograded questions will result in updated autogrades and updated graded score, just as if the attempt were in the In Progress state initially.
- Reset all grading rubrics by clearing them to their pre-instructor-grading state.
- Do not clear instructor evaluation comments within LRM content. We leave the comments so instructors can refer to them when they return to grading after resubmission.

DCR: Clearning LRM Instructor Comments During Reactivate

If this becomes confusing because instructors don't know which comments are from the previous attempt and which comments are from the reactivated attempt, a couple options are:

- Delete the instructor comments too. The value in keeping them depends on how instructors intend to use this feature. It may be that they expect all changes they made to be cleared.

- Could add a prefix to the instructor's comments [Previous Scoring: 17.5] <Instructor Comments>. The ": <score>" would only display if the instructor had specified a score during the previous attempt. The intent of the [Previous Scoring] flag is to show what comments were applied before reactivating to the learner.

SCORM 1.2
Since there is no instructor grading allowed in SCORM 1.2, we don't need to reset any evaluation scores. The graded score should always reflect the current autograded state.

IMS+
IMS+ is a combination of LRM and SCORM 1.2 activities. We follow the steps outlined above on an activity-by-activity basis. This should result in a scoring reset to the pre-submitted autograding state. As learners proceed through the content and make changes, the autograding and thus graded score will update accordingly.

SCORM 2004
The same activity tree attempt is used and the learner continues the attempt from the beginning of the activity tree, as if they are starting the attempt. See section 7.2 for design history on SCORM 2004 reactivation.

6.2. Scoring Model

Scoring is the process of reviewing and setting points for a learner assignment. Scoring occurs on the grading page, where the instructor can select learner assignments to grade and can provide final points and overall comments in the grading chart.

There are two main concepts in the scoring model:

- Automatic and manual scoring *inside* the e-learning content is used to calculate the *Computed Points* value for a learner assignment.
- Instructors set *Final Points* for the learner assignment in the grading chart. Final points is initialized to computed points, but an instructor can override the final value. Final points is considered the final score for the learner assignment.

Scoring occurs after a learner assignment is submitted as complete. An instructor may not edit final points or perform any in-content manual scoring until the learner assignment state is "completed".

The example below gives some context for how computed and final points are used in SLK. The sub-sections that follow describe specific details of the scoring model.

Example
This is an example of how the scoring model works for Class Server content (LRM) in SLK. 1. An Instructor, Bob, creates an assignment and assigns content to a learner, Joan. 2. While Joan's learner assignment is in the "not started" or "active" states, Bob cannot access grading view or set the final points value for Joan's learner assignment. 3. When Joan submits her assignment, or if Bob "collects" her learner assignment, computed points is calculated and final points is initialized to the computed points value. Bob can now see computed points in the grading chart and the final points field is enabled for editing. (Computed Points = 5, Final Points = 5) 4. Bob can edit the final points value to give Joan 2 extra credit points. (Computed Points = 5, Final Points = 7) 5. Bob can click on Joan's name in the grading chart to open her learner assignment in grading view. He changes the interaction score for the second question from 3 to 4 and also enters 5 points for a non-auto-graded question score. 6. After closing the frameset, Bob notices that computed points in the grading chart is adjusted to reflect his changes. He also notices that the final points column was modified by the same delta. (Computed Points = 11, Final Points = 13)

6.2.1. Points

Points is the normalized unit of measure displayed by SLK to represent the "score" of a learner assignment.

The internal "automatic" scoring calculations differ by assigned content type, so SLK normalizes those scores to *points* for display at the Assignment or Learner Assignment level. This is done so we can have a more normalized "scale" of points for SLK rather than having some assignments with scores ranging from -1 to 1 and other scores ranging from 0 to 100.

6.2.2. Points Possible

This is the nominal maximum number of points possible for a given *assignment*. It is possible for a learner to be awarded more points than this value (e.g. extra credit).

Points possible is set when an instructor creates or edits an assignment. It defaults to the values in the table below. An instructor may edit the value at anytime, including after an assignment is created. Points possible may be null.

Content	**Default**	**Where Stored**
LRM	Default from index.xml	AssignmentItem.PointsPossible
SCORM 2004	100	AssignmentItem.PointsPossible
SCORM 1.2	Blank	AssignmentItem.PointsPossible
Non-e-learning	Blank	AssignmentItem.PointsPossible

The Assignment List Web Part and SLK Assignment Workflow UI display points possible when showing the user's score for an assignment:

<learner assignment final points>/<assignment points possible>
Example: 12/25

6.2.3. Per-Attempt Computed Points

Computed Points is the SLK normalized points value calculated from internal scoring of the e-learning content. This includes content auto-scoring and, for LRM, any in-content manual scoring done by the instructor. Computed points may be null.

Computed points is first set when the assignment is submitted as complete. Non-e-learning content does not have auto-grading or computed points.

Computed Points is calculated as follows and stored in AttemptItem.TotalPoints:

Content	Computed Points Calculation
LRM	When first calculated: • All auto-graded questions are graded. • For each auto-graded question, the question's instructor score is initialized to the auto-graded score. • Each auto-grade score is stored separately in the database for reporting purposes. • The instructor score for non-auto-graded questions is initialized to null. Computed points is: $\sum$ (Instructor score for each interaction of each LRM activity) For LRM packages that also contain SCORM content (IMS+), the SCORM content is ignored for the computed points calculation.
SCORM 2004	Scaled score of the primary objective of the root activity * 100
SCORM 1.2	$\sum$ (Raw score for each interaction of each activity)
Non-E-Learning	Null

6.2.3.1. Per-Interaction Auto-score (LRM)

Each autogradable interaction in LRM has an auto-score. LRM auto-scores may be computed at any time, but must be available when the learner assignment is in the *completed* or *final* state. SLK uses the per-interaction auto-score as the default value for the per-interaction instructor scores. The per-interaction auto-score is stored for reporting purposes. This value may not change after the learner assignment has been *completed* and autograded, unless the assignment is reactivated.

Autograded questions that are unanswered have an auto-score of zero. The corresponding instructor score is initialized to the same value.

6.2.3.2. Per-Interaction Instructor Score (LRM)

For each interaction in LRM, the instructor may set an instructor score. For autograded interactions, this score is initialized to the per-interaction auto-score. If the interaction is not autograded, then the instructor score is initially blank. The instructor may edit this score in grading view.

6.2.3.3. Non-e-learning content

Non-e-learning content is not autograded and does not have a computed score. Thus the final score initializes to blank.

6.2.4. Learner Assignment Final Points

Final Points is "the" score for a learner assignment. This value is initialized to the value of computed points, but may be edited by the instructor when the learner assignment is in the "completed" or "final" states.

The instructor may clear the final points to indicate that no score should be recorded for the learner assignment. This is most common when the user is excused from the assignment.

If the computed points value changes, then final points will change by the same delta as defined in section 6.2.5.

The learner should not be able to access final points unless the assignment state is "final". In all other states, the score should be displayed to the learner as if it is null.

6.2.5. Changes to Computed Points and Final Points

Changes to *Computed Points* update *Final Points* by the same delta change as follows:

- If computed points was previously null, then final points wasn't empty because of an instructor's modification and should therefore be updated with the new computed points.

```
if (computedPointsOld == Null) {
        finalPointsNew=computedPointsNew}
```

- If there was already a computed points value, then if final points isn't null, update it with the change in points. If there was a computed points value and final points is null, then it was cleared by the instructor and shouldn't be modified by the change in computed points.

```
else {
if (finalPointsOld != Null)
        finalPointsNew=finalPointsOld+(computedPointsNew-computedPointsOld)}
```

General information on how computed points changes final points:

- The points in the grading chart are updated via inter-window communication when the frameset is closed from grading view.
- The instructor can update the grading chart while the frameset is open in grading view.
- Each save of the frameset will store the latest computed and final points value on the server.
- Saving the grading chart will store the current final points displayed in the chart.
- Computed points is only modified by the content scoring and isn't modifiable by the grading chart. This makes it a good "master points" reference if final points is out of synch and must be updated by the instructor.

6.2.6. SCORM 2004 Success Status

SCORM 2004 e-learning content has the concept of success status that is separate from score. Success status is determined by the content and e-learning run-time and is not modified or set manually by the instructor.

Success status is the success status of the primary objective of the root activity. The Assignment Workflow UI specification defines the UI for displaying success status.

7. FAQs

7.1. SLK Authorization

Using roles rather than groups
In early designs, we planed to use groups rather than roles to specify the learners and instructors in SLK. In WSS V3 the concept of site-group has been removed and replaced by roles. You may no longer create a group at the SPWeb level with unique permissions to that SPWeb. In V3 groups exist at the SPSite level and are shared across SPWebs, but different roles may be applied to those groups at the SPWeb level by breaking the role inheritance. Typical school site hierarchies have a school SPSite and separate SPWebs and sub-SPWebs for each class or course. We need to allow users to specify unique sets of learners and instructors at the SPWeb level. For example, when creating an assignment on the History6 SPWeb, you will only want to assign to the relevant users and groups in the scope of History6. You will not want to see all users and groups in the entire school. Using groups alone to do this would require a forced group provisioning structure and naming convention to match the group name with the appropriate SPWeb because groups are shared across a SPSite.

Rather than using groups, we see roles as the proper way to identify types of users (SLK Learner or SLK Instructor) in the scope of a given SPWeb. Roles also provide greater flexibility in the end implementation. There may be several groups assigned the SLK Learners role for a SPWeb, or groups may not be used at all and individual users may be assigned the role.

Why not use a default role rather than recommending SLK Learners and SLK Instructors roles?
We recommend using SLK Learners and SLK Instructors so you can apply a role to users and groups for identification in SLK without affecting other permissions used by SharePoint. An administrator could choose to use a default role (Full Control, Design, Contribute, Read), but it is possible that these roles are applied to more users than desired and lead to too many learners or instructors being identified or the wrong learners and instructors being identified.

Using SLK Learners and SLK Instructors lets the SPWeb administrator have full control of who has those roles in SLK and does not require them to change their existing SharePoint permissions structure.

Why do we use a permission to identify learners in SLK rather than just allowing all members of a site to be learners?
The issue is in determining who "all the members" of a SPWeb are. Users and groups are scoped to the SPSite level, but the assignment of permissions to those users and groups can be scoped to the SPWeb level. We want to be able to be able to identify learners for SLK at the SPWeb level. This allows instructors, for example, to create a new assignment and only list the users with the SLK Learner role for the given SPWeb as available learners for the assignment. One alternative would be to allow all users and groups with any permission assigned on the site, but using a specific permission allows the instructor more control on which users and groups are allowed to create assignments (SLK instructor permission) and which users and groups are listed as available learners in an assignment (SLK learner permission).

Note, the actual learners and instructors on any given assignment are stored in the SLK database. The SharePoint permissions are used when determining the user availability when creating/editing

in the assignment properties page and when verifying a user is an instructor on the target SPWeb for creating an assignment.

It is possible that an administrator would like to use an existing permission like "contribute" to leverage already created permission assignments. For example, by default the "<SPWeb Name> Members" group is created when an SPWeb is created and it is given the "contribute" permission for the SPWeb. In this case the user could specify the "contribute" permission to be used as the identifying permission for learners in SLK. This is why we allow administrators to specify what permission should be used to identify learners and instructors in SLK for an SPSite.

7.2. SCORM 2004 Reactivation

The original design for SCORM 2004 reactivation created a new attempt for the learner to execute. While this would delete all of the learner's previous work, the thought was it would give them a clean sequencing data model to work with because there are possible situations where the sequencing rules will not allow the user to work through the entire assignment again and could cause the content to exit automatically upon re-entering.

In evaluating this design, it didn't fully fix the problem. Content will often use global objectives to retain information for sequencing decisions. These objectives are global to the learner and package and can be global to the learner and the LMS if globalToSystem is true. This means that even with a clean attempt the learner may still not get fresh sequencing.

Because the design doesn't fully fix the potential sequencing issues, we have decided to keep the reactivate behavior for SCORM 2004 the same as other e-learning content types. This will maintain a consistent behavior between content types and will eliminate the data loss associated with creating a new attempt on the activity tree.

For situations where the original design (new attempt) would have allowed the learner to work in the assignment and the implemented design does not, the instructor can still create a new attempt for the learner by removing and re-adding them to the assignment.

8. Proposed Design Changes

9. Spec History

Date	Changes Made
5/10/2006	Spec created
5/15/2006	• Incorporated the following core concepts from other specs. ♦ SLK Authorization ♦ Assignments and LearnerAssignments ♦ LearnerAssignment state and state calculation ♦ Scoring model • Added "SharePoint Roles vs SLK Assignment Relationships" to describe the difference between, for example, the learner SharePoint role and being a learner on a particular assignment. • Added "Assigning to Individuals, Selecting by Groups" to describe that assignment

	creators are able to select learners either individually or by group, but in both cases learner assignments are only created for individual users.
5/30/2006	• Added description of caching and learner access to cached packages. • Added some terminology • Made quite a few changes to the scoring model text. Reorganized some of the sections to improve flow. Changed that computed points updates in the grading chart occur on frameset close from grading view rather than on every save of the frameset.
6/06/2006	• Changed the SLK/MLC Views chart to show that the SLK Grading View will use the MLC Random Access view. Previously we planned to have a different MLC grading view, but the main difference in functionality was that grading view only access leaf nodes, where Random Access can access all nodes. SLK will use random access and just not use the non-leaf-node capability. See the SLK Frameset spec FAQ section for additional information. • Package cache location and expiration time will be defined in SLK settings and thus set per site collection. By default, the settings file does not specify the cache location and packages are cached in the temp directory of each FE server. The default cache expiration time is 3 days. • Added State Change Behavior section to describe what e-learning attempt actions and state changes occur for different workflow actions like submitting an assignment.
6/27/2006	Summary • Added brief summary description of SLK • Removed background/system requirements section because it is covered in other documents. SLK Installation and Configuration • Added SLK Installation and Configuration section to describe the basics of SLK as a WSS solution with integrated SharePoint configuration. Provisioning and Authorization • Updated site hierarchy diagram and description to describe the parallels of SLK with SharePoint. • Minor text cleanup throughout the section. • Set placeholder for LearningStore Views and Security. Placeholder text in the comment. Schema, Views, and Queries • Added SLK Query and Query set background information. • Modified LearningStore Views vs SLK Queries section to describe the basic differences and how SLK queries use LearningStore views. • Have place holders for sections on e-learning content. Assignment Workflow • Updated Assignment-to-LearnerAssignment relationship diagram. • Removed links to specs and internal pages. • Modified "Instructor is also a learner" section to clarify that the workflow UI spec specifies what views are available in this situation. • Removed any references to reactivate warning UI. All reactivate UI is specified in the SLK Assignment Workflow UI spec. • Removed internal links to old Class Server reactivate specs. • Removed Instructor-led and self-directed workflow sections. If they are needed later, we can add them back in. • Removed range and precision columns in the points possible table because we eliminated the application range constraint of -10000-10000. This was a carry over from CS 4 that we no longer need. • Removed "Where Stored" for the Computed Points calculation table. This is left to the implementation. • Removed IMS+ section in per-attempt computed points section because it is covered in the sections discussion LRM. • Moved Instructor Review DCRs to the SLK Frameset specification.

	• Removed corrupt table in Learner Assignment Final Points section. • Removed example on modifying final points in the grading chart while in grading view for LRM content. The previous example was based on a design where there could be continuous updates throughout editing. Current design updates the grading chart when the frameset closes. Error Conditions • Removed this section. Errors are covered in each individual spec. General • Completed spec cleanup for CTP1 release.
6/28/2006	Minor changes from review for CTP1.
8/3/2006	SLK Authorization • Added section on 'Authorization During Assignment Creation' to clarify the permission requirements when creating an assignment. LearnerAssignmentStates • When the 'autoreturn' property is set for an assignment, any completed learner assignments are advanced to final the next time they are accessed. • Fixed 'Reactivating a Learner Assignment' section to more clearly define the reactivate behavior for e-learning content. Scoring Model • Computed points is stored in AttemptItem.TotalPoints. • SCORM 2004 computed points is scaled score of the primary objective of the root activity*100. • SCORM 2004 success status is the success status of the primary objective of the root activity. I now reference the workflow UI spec for details of how this is presented in the UI and removed any UI descriptions (pass/fail, etc.).
8/31/2006	Added DCR for logging learner assignment state changes.
9/29/2006	Instructor and Learner Permissions • Instructor and learner permissions are no longer assigned rights during creation. SLK uses the permissions to identify instructors and learners, but the SLK application now uses the app pool account to access necessary information from SharePoint rather than requiring a user to have the higher rights at all times. Reactivating Assignments • Changed reactivate behavior for SCORM 2004 content. It will now maintain the same attempt rather than creating a new attempt for the learner. • Reactivating LRM content will reset the instructor evaluation scores to the current autograded scores. This will basically reset the content and scoring back to the in progress state. Instructor comments in the content will be maintained. Learner assignment comments are maintained, but final points is cleared.
10/24/2006	General Updates • Described domain group behavior and the core concepts of assigning permissions to domain groups individually or by adding them to a SharePoint group with the permissions.(Bug #708) • Minor edits to update text. • Added package caching note that assignments are created for a particular version of a file and any changes to that file will result in an error displayed to the user. (Bug #563)
12/7/2006	**Queries and Views** • Removed empty sections on LearningStore security and the schema. This is covered in the SDK documentation. **Supported Content Formats** • Added note that SLK does not support Class Server remote content. (Bug #931)

	• Cleaned up content section and filled out basic information about the content types and their versions supported by SLK. **General** • Removed hyperlinks to internal servers. (Bug #1321)