Liferay Objects

Lecture Script

# Introduction

Slide 1: [Read out Slide Numbers]

Hello, my name is [your name]. In this module, we’ll be learning about Liferay Objects, a new feature introduced in Liferay DXP 7.4 *[seven dot four]*.

Slide 2:

Our Key Module Takeaways for this section are as follows:

Liferay Objects provides a low-code way to create custom business objects through the Liferay DXP UI.

Objects define the information an application collects and manages. Non-developers can define fields, layouts, and relationships for Objects.

Objects can be integrated into Liferay Workflow, Permission, and Asset Frameworks for improved out of the box usability.

Objects are deployed as Widgets and entries displayed as collections.

Slide 3:

Before we begin, make sure you have a Java JDK installed to run Liferay DXP. This can be done using the link on the screen [at oracle.com], which will direct you to download Java JDK 8. Note that using more recent versions of Java JDK may result in errors.

Also, be sure that you have unzipped the module exercise files in the appropriate folder structure for your OS. For Windows, we recommend the C:\Liferay path. For Unix systems, we recommend the [user-home]/liferay path.

Slide 4:

Our use case for this module is the Mondego Group. The Mondego Group is an international financial services organization that is using Liferay DXP to create its customer service and employee portals.

The Mondego Group needs to collect and store information submitted by customers when they open new accounts. Mondego can use Liferay Objects to: Manage how information is collected and stored in Liferay DXP, Create custom forms and workflows integrated with Objects, and Define relationships between business objects, such as New Accounts and Account Types.

**Slide 5:**

Mondego has the following Key Performance Indicators. The company wants to:

Improve efficiency of processing applications by integrating Objects with Workflow

Reduce the collection of extraneous user data

Decrease the amount of time needed to create custom business objects and applications

*(End Video One)*

# Creating New Objects

Slide 6:

In this section, we’ll discuss how to create Objects in Liferay DXP 7.4.

Slide 7:

Liferay Objects, introduced in Liferay DXP 7.4, is a low-code way to build and deliver applications without writing code or deploying modules. The Objects application allows non-developers to quickly develop custom solutions for their business needs.

Objects define what information an application collects and manages. This is done by adding custom data fields, defining complex relationships, designing custom layouts, and defining Object scope.

From creation, all Objects are fully integrated with Liferay's core frameworks.

Our use case example, the Mondego Group, can use Liferay Objects to quickly create applications to use across the platform. The team can create a custom New Account Object integrated with Liferay Forms that allows customers to easily fill out applications for new accounts with Mondego.

Slide 8:

In Liferay DXP, Objects are created using the Objects portlet. The portlet is located in the Control Panel. To create a new Object, you first provide a Name and Plural Name. The name will generate . First, when you add a new Object, provide a Name, Plural Name, and Object Name. The Object Name generates a Table Name which cannot be changed Once saved. Once this is complete, the Object opens as a draft.

While the Object is in draft form, you can edit it in a variety of ways. You can change the Object Name and Labels and add or delete fields, relationships, layouts, and actions. You can also define the Object Scope. And, if needed, Objects in draft state can be deleted.

Object Scope refers to how data is stored. Company, the default, means that data is stored per Liferay instance. Site scoped Objects store data per Site. These scopes also determine where an Object can be listed in the menu. Company scoped Objects can be displayed in the Global Menu while Site scoped Objects can be listed in the Site Administration Menu.

Before you publish a draft be sure to check the Scope, Name, and any added fields or relationships to verify that they are correct. Once a draft is published, these elements cannot be changed or deleted, and the Object itself cannot be deleted.

When the Mondego team creates the New Account Object, they add several fields and one relationship between New Account and Account Types. Additionally, they scope the Object to Site so that each banking Site stores a unique dataset of New Account submissions rather than all New Account submissions being saved at the instance level.

Slide 9:

Fields are a key component of Objects. Fields are data definitions that represent a database column and store different types of values for Liferay Objects. Remember, Objects generate an actual table in an instance’s database.

Field Types include:

BigDecimal, which is a high precision numerical value with a decimal point used for calculations.

Boolean, a logical, binary value of either true or false.

Date, a value indicating a specific day, month, and year.

Double, a 64-bit numerical value with a floating point

Integer, a 32-bit numerical value *without* a floating point

Long, a 64-bit numerical value *without* a floating point

Picklist, which is a string value stored in a picklist. We’ll discuss these a little later.

And, finally, String, which is a sequence of characters.

Fields can be added to the Object draft or after the Object has been published. Additionally, Fields can be set to be mandatory or optional and searchable or not.

So, for Mondego’s New Account Object, a variety of field types can be used. The String Field is used for Names, the Date field for Date of Birth, the Boolean for a Yes or No question about whether the customer has ever made an account with Mondego before, and the Picklist to provide a list of options for Account Types.

Slide 10:

Relationships can also be added to a draft or published Object. Relationships are connections between Objects. Liferay Objects provides two types of relationships: One to Many and Many to Many.

One to Many Relationships mean that one of the current Object’s entities can be related to multiple entities in another Object. The table generated for this relationship will add the entity representing the One (or parent) to the Many (or child) side.

For example, Mondego’s New Account Object needs to be connected to the Account Type Object. The ideal relationship for this is a One to Many: One Account Type to Many New Accounts. When this relationship is created, the Account Type entity will show up as a new field in the New Account Object.

Many to Many Relationships mean that multiple entities in the current Object can be related to multiple entities in another Object. In this case, a new database table is created for both sides and can be added to a Relationships Tab in either Object’s custom layout.

Keep in mind that you must create a custom layout to display relationships for Object entries.

Slide 11:

Object Layouts define how fields and relationships are organized and displayed when creating or editing an entry. The default layout for Objects is one tab with all the fields displayed in alphabetical order, but custom layouts allow you to add multiple tabs and manage field and relationship organization.

There are two types of tabs in Layouts: Fields and Relationships. Field tabs are used to display Object entry fields. Relationship tabs are used to display relationship tables. For One to Many relationships, the tab is used to display the parent side, whereas the tab displays both sides in Many to Many relationships.

Rather than displaying all fields in alphabetical order on one page, the default, the web team at Mondego creates a new default display layout for the New Account Object. The first tab asks for personal information and the second tab focuses on account-specific details.

Keep in mind that Relationship tabs can only be added after a Field tab has been created.

Slide 12:

*Picklists* is an application for creating standard lists of values that are understood across the Liferay Portal. By adding items to a list, that list can then be used in supported applications, saving time and effort.

Remember that one of our Field Types in Liferay Objects is a Picklist. So, on the Mondego platform, a Picklist is created for Departments and then added as a Field for the PTO Requests Object. Rather than requiring each employee to manually enter their department when they submit entries for PTO Requests, the picklist field provides a dropdown list containing all available options (e.g. Human Resources, Marketing, Engineering, etc.). Employees can simply select the correct department.

Once a list is used in an Object field, the list cannot be deleted. However, list items can be edited and removed. Updating or deleting a Picklist item will automatically update all Object entries using the item.

Slide 13:

When you’re ready to publish an Object, the following operations take place. A database table is created for the Object with the data definitions. A new Headless API is created for CRUD operations. A Collection Provider is created for displaying Object entries. And, the Object is integrated with Information, Permissions, Workflow, and Form frameworks

Once an Object is published it cannot be deleted, the Name and Scope cannot be changed, and any Relationships and Fields created before publication cannot be removed or reconfigured. So, it is important to check the Name, Scope, Fields, and other elements to make sure they are correct before publishing.

Although Objects cannot be deleted once published, they can be deactivated to hide them from Users.

So, once the New Account Object is ready, the team can hit publish and begin using it on various Sites across the Mondego Platform.

Slide 14:

Let’s review our Use Case for this Section. The Mondego Group’s Banking Sites plan to use Objects to create custom applications for New Account submissions. When new customers open a Mondego Bank Account, the information added to the New Account Object will automatically be saved to the respective database table. By creating a One to Many relationship between the Account Type Object and the New Account Object, a field for Account Type will automatically be added to the New Account Object.

Mondego can also create a custom Object for internal Requests. When an employee needs to make a request for PTO, their information is collected in the database table for PTO requests. For this Object, they improve organization of the data by creating a Picklist for Departments so employees can select their department when they submit PTO requests.

Slide 15:

Now, we’ve reached our Knowledge Check for this section.

(Blank) allows Mondego to create custom applications without writing code.

When creating a New Account Object, different (Blank) can be added with each storing a different type of value, including (Blank), (Blank), (Blank), (Blank), (Blank), and others.

Mondego can create connections between different objects using Relationships:

(Blank)

(Blank)

Mondego can create a Departments (Blank) to store a standard list of values that can be used in supported applications.

*(End Video Two)*

# Integrating and Displaying Liferay Objects

Slide 23:

In this section, we’ll discuss how Liferay Objects are integrated and displayed across the Liferay Platform.

Slide 24:

By default, all Objects are fully integrated with Liferay's core frameworks from creation, including Permissions, Workflows, Forms, Assets, and Headless REST APIs.

This full integration of Objects with Liferay’s core frameworks allows you to provide a unified experience across the platform.

Slide 25:

Since Objects are automatically integrated with the Permissions framework, you can assign application and resource permissions to manage access to the Objects and their entries. Permissions for Objects include: access to the Object in the Control Panel or Site Menu; permission to add Object entries; and permission to update, delete, and view entries in Objects.

Permissions for Objects are split into two categories, Application Permissions and Resource Permissions. Application Permissions include Access in Control Panel, Configuration, Permissions, Preferences, and View.

Resource Permissions include two subcategories. The *Resource-Related Actions* subcategory includes add object definition, extend system object definition, permissions, and publish object definition. The *Actions on Database Entities* subcategory includes delete, permissions, update, view.

As with all permissions in Liferay DXP, permissions for Objects must be defined for a particular Role, and then the Role must be assigned to a User to take effect. This ensures that only appropriate users can view and use an application's data.

The Mondego team can create Roles with set permissions to determine who can create, manage, and add to Objects. For example, entries can only be made to the Object Account Type by Site Administrators whereas New Account entries can be made by Site Users.

Slide 26:

Once again, since Objects are automatically integrated with the Workflow framework, adding workflow steps to Object entry is simply a matter of setting the desired workflow.

Object entry creation can be set to follow a defined review and approval process using Workflows. By default, Liferay DXP offers either No Workflow or Single Approver Workflow, but additional Workflows can be created to better fit certain Objects.

For instance, if we set the Object Entry workflow to Single Approver, it will require that an Administrator review and approve of the new entry before it is officially added. This is especially important for our use case. The Mondego Group wants to make sure all New Account applications are valid entries, so they implement a Workflow that requires all submissions to be approved by a Site Administrator.

Slide 27:

Integrating Objects with Forms allows you to collect data using a Form and store that information in the Object database.

By default, Forms save entries in a separate Entries tab next to the Form and Rules tabs. By setting an Object as the Storage Type, you can match the form fields to the Object fields. This then saves the entries to the appropriate fields in the Object database instead of in the Entries tab.

Once the Form is created, a shareable link is generated which can then be used to collect information.

Mondego Banking has an Object for New Accounts. By integrating that Object with Forms, they can create a Forms link that anyone can access. Any responses to that Form will automatically be saved in the New Accounts Object, which can be accessed in the Site Menu.

Slide 28:

Object entries can be displayed using the Collection Display fragment, which has two main layouts, Grid and Table. You can manage these layouts and select the Object entries you want to display under the Collections section of the General Tab. Available Objects are listed under the Collection Providers tab.

To display each individual entry in a Collection Display as its own Page, you can use Display Page Templates. To do so, simply set the Content Type of the Display Page Template to the desired Object. Then determine the layout you want and set it as the template for the Object.

Using a Collection Display fragment, the Mondego can quickly display all Object Entries on a page. For example, they want to include an overview of all recent applicants on the private page “Applicants.” To make each entry appear on its own page in case they need to concentrate on an individual, the web team creates a Display Page Template for New Accounts which turns each entry into a link that redirects to a page with only that individual’s information.

Slide 31:

Let’s review our use case once again. By taking advantage of the automatic integration of Objects with Liferay’s core Frameworks, Mondego can:

Create a Form integrated with the New Account Object so users can easily submit applications to open new accounts.

Add a Workflow to New Account entries, requiring all new entries to be reviewed by two separate reviewers before the submissions are approved or rejected.

And, Display all New Account entries on a private page on the Mondego Site using the Collection Display fragment and create links to view each unique entry by applying a Display Page Template

Slide 32:

Now, we’ve reached our Knowledge Check for this section.

Liferay Objects are fully integrated with Liferay’s (Blank), so Mondego can immediately leverage those capabilities, including:

Managing who has access to Object entries with (Blank)

Assigning a (Blank) to entries so that changes are reviewed before they are published

Mondego can set (Blank) to store data in an Object so that new entries can be added by general users and still be saved to the Object database.

To display Object entries, Mondego creates a new Content Page with a (Blank) fragment. Applying a (Blank) to the Object allows each entry in the table to link to a distinct page.

*(End Video Three)*

# Summary

Slide 46:

We’ve now reached the end of our module, so let’s review.

Liferay Objects is a tool that businesses can use to build and deliver custom, low-code applications within the Liferay DXP UI.

Objects are created using the Objects portlet where a draft is created, assigned fields, scopes, and relationships, and finally published, which creates the new application within the platform.

Fields are data definitions that represent a database column and store different types of values for Liferay Objects. They operate similarly to the fields found in the Forms application.

Slide 47:

Relationships are connections between Objects and can be either One to Many or Many to Many.

Object Layouts define how fields and relationships are organized and displayed and can be presented in either single or multi-tab arrangements.

Upon publication, the following takes place for a new Object: a database table is created for the Object with the data definitions, a new Headless API is created for CRUD operations, a Collection Provider is created for displaying Object entries, and the Object is integrated with the Info, Permissions, Workflow, and Form frameworks.

Slide 48:

Once created, Objects cannot be deleted or changed in name or scope.

From the time of their creation, all Objects are fully integrated with Liferay’s core frameworks including permissions, workflows, Assets, and Headless APIs.

Forms can be used to easily collect and store entries in an Object’s database.

Object entries can be displayed collectively using a Collection Display Fragment or individually using Display Page Templates.

Slide 49:

For more information about Liferay Objects, you can consult the documentation listed on the screen:

https://learn.liferay.com/dxp/latest/en/developing-applications/developing-low-code-applications/objects/introduction-to-objects.html

Or, for general questions about using Liferay’s products, visit learn-dot-Liferay-dot-com.

https://learn.liferay.com/index.html

*(End Video Four)*