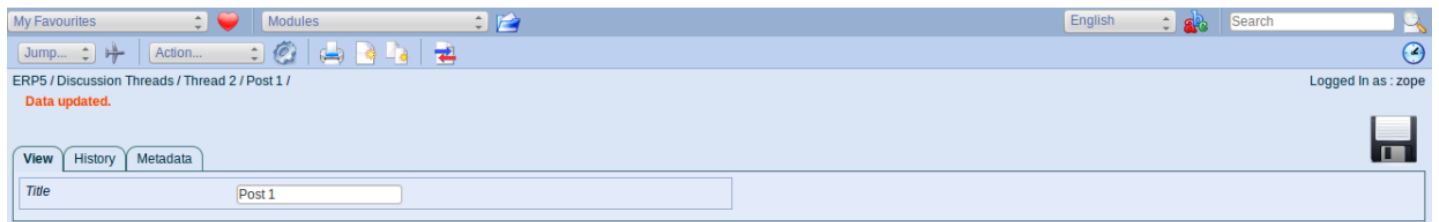


How To Create A New Portal Type

Example: Discussion Post -a new Content type in Discussion Threads Module



▼ Details

This tutorial explains how to create a new portal type in ERP5. In our example we will create the Discussion Post type-a new Content type in Discussion Threads Module.

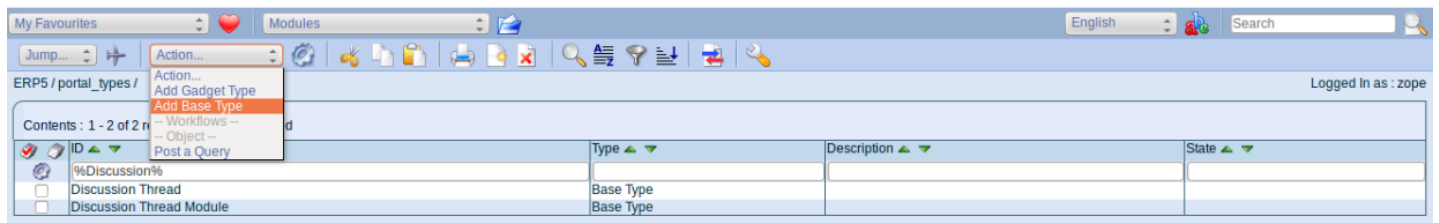
Agenda

- How to create a Discussion Post type
- How to allow Content types
- How to create Threads and Posts

▼ Details

Add Base Type

INSTANCE_ID/portal_types/view



▼ Details

We have seen previously how to create a new module. It is now time to see how to create the new portal type which can be contained by the module.

You must go to the portal type list, in order to access that first put your instance ID and then /portal_types/view like it's written on the slide. You will then be able to see all portal types which have already been created in your ERP5 instance. They are used by the modules which are installed on your instance.

If you filter the list of available Portal Types by putting %Discussion% in the ID listbox, you will see the previously created documents. We have the Discussion Thread Module, which is a folder, then a Discussion Thread which is an ERP5 default document. We need to create the "Post" portal type.

Once you are on this list, you can create a new Base Type. Click on "Action..." menu and choose "Add Base Type".

Edit portal type

My Favourites Modules English Search

ERP5 / portal_types / Discussion Post /

Data updated.

View Actions Roles Translations History Metadata

ID Discussion Post

Short Title

Class XMLObject

Init Script ID

Mixins

Interfaces

Add Permission

Acquire Local Roles

Property Sheets XMLObject

Base Categories

Icon

Filter Content Types

Allowed Content Types

Hidden Content Types

Groups

Searchable Text Property IDs

title
description
reference
short_title

Searchable Text Method IDs

Description A post for a forum

▼ Details

Once the portal type has been created, you must now set up its properties. The **Base Type ID** is "Discussion Post". The **Class** is the class that your Discussion Post will inherit from. In our case we will use "XMLObject" which stands for ERP5 base XML objects. The **Property Sheets** is the property which will define the type of the object, and give it all its default values and properties. In our case we will select "XML object". If required we could have add more property sheets.

Once the changes have been made don't forget to save them thanks to the floppy disk icon.

Create View action for the object to be rendered

My Favourites Modules English Search

ERP5 / portal_types / Discussion Post /

Data updated.

View Actions Roles Translations History Metadata

ID Discussion Post

Short Title

Class XMLObject

Init Script ID

Action...
Add Action Information
Add Role Information
-- Workflows --
-- Object --
Update Local Roles
Post a Query

My Favourites Modules English Search

ERP5 / portal_types / Discussion Post / View /

Data updated.

View History Metadata

Reference view

Title view

Action string:\${object_url}/DiscussionPost_view

Icon

Condition

Permission View

Category object_view

Visible

Priority 1.0

Description

▼ Details

We now have to create an action for the object to be rendered. Create an action named "View" by clicking on the menu "Actions..." and then "Add action information". It's the default action which is called when the object has to be shown on a page.

Here the two important values are Action and Permission. Action defines what form will be used to render the object and Permission defines what permission will be associated with this action. Here we use `string:${object_url}/DiscussionPost_view` as Action. Also set Category to `object_view`.

When this action is added, all the Discussion Post object acquire a method View. This method is the one which is called when you try to display the object with a web browser. With these settings, ERP5 will call the `DiscussionPost_view` form in a tab call View if you put the "View" permission on the action.

We now need to create the `DiscussionPost_view` form, otherwise ERP5 will raise an error when we will try to view a Discussion Post object.

Create the view well-rendered

INSTANCE_ID/portal_skins/erp5_forum/manage_main

The image shows two screenshots of the ERP5 file manager interface. The top screenshot shows a folder named `/erp5/portal_skins/erp5_forum` containing three files: `DiscussionThreadModule_viewDiscussionThreadList (Discussion Threads)`, `DiscussionThread_view (Discussion Thread)`, and `copy_of_DiscussionThread_view (Discussion Thread)`. A context menu is open over the `copy_of_DiscussionThread_view` file, with the `Rename` option selected. A red arrow labeled '1' points to the `Rename` button. The bottom screenshot shows the same folder, but the `DiscussionThread_view` file has been renamed to `DiscussionPost_view (Discussion Thread)`. A red arrow labeled '2' points to the `DiscussionPost_view` text in the file list. A 'Rename Items' dialog box is also visible in the top screenshot, showing the file name being changed to `DiscussionPost_view`.

▼ Details

In order to create the `DiscussionPost_view` form, we have to access to Portal Skin Folder of the Discussion Threads module. Every code about how to render the elements of your module is contained in the Portal Skin Folder of this module. If you remember the Portal Skin Folder associated with the Discussion Threads module that you created before was named `erp5_forum`. You should access it through a URL like `INSTANCE_ID/portal_skins/erp5_forum/manage_main` (`manage_main` allows you to access the zope management interface - ZMI).

This folder contains all the elements of your module. You can see that ERP5 has put here the views of your Discussion Thread and Discussion Thread Module objects: they were automatically created when you created the module. You can put many other things such as scripts, templates, CSS, images, etc.

In our case we need another view form, very similar to the one which have been created for the Discussion Thread. This is why the fastest way to create the view is to copy/paste the existing view.

Copy/paste the `DiscussionThread_view` and rename it to `DiscussionPost_view`.

Edit the view renderer

ERP5 Form at /erp5/portal_skins/erp5_forum/DiscussionPost_view

Settings for this form.

Title 2 Discussion Post

Description

Number of groups in row (In order tab) 4

Form name 3 DiscussionPost_view

Page Template form_view

Form action Base_edit

Form update action

Update Action Title

Form method POST

Form enctype No enctype

Encoding of pages the form is in UTF-8

Encoding of form properties UTF-8

Form properties are unicode

Setters for these properties should be called by edit() in the defined order

Change 4

Upgrade

Upgrade

▼ Details

Once the form has been renamed, we must change some settings.

You can easily do the basic operations on each object if you access it by the /manage method. Every object in erp5 can be managed. For the basic objects you will be able to add/delete/modify contents of this object and for advanced object such as workflows, there will be special actions to modify them.

In ERP5 titles are for "User Only" usage, you can't use titles in any script because it's just a rendering option (see coding crimes). You can correct the title of the form to "Discussion Post".

Carefully correct the Form Name and the Title in Settings tab of the DiscussionPost_view form and click on "Change".

Allowed Content Types

ERP5 / portal_types / Discussion Thread /

Data updated.

Logged In as : zope

View Actions Roles Translations History Metadata

ID Discussion Thread

Short Title

Class XMLObject

Init Script ID

Mixins

Interfaces

Add Permission

Acquire Local Roles

Property Sheets

Base Categories

Icon

Filter Content Types

Allowed Content Types Discussion Post

Hidden Content Types

Groups

Searchable Text Property IDs title description reference short_title

Searchable Text Method IDs

Description

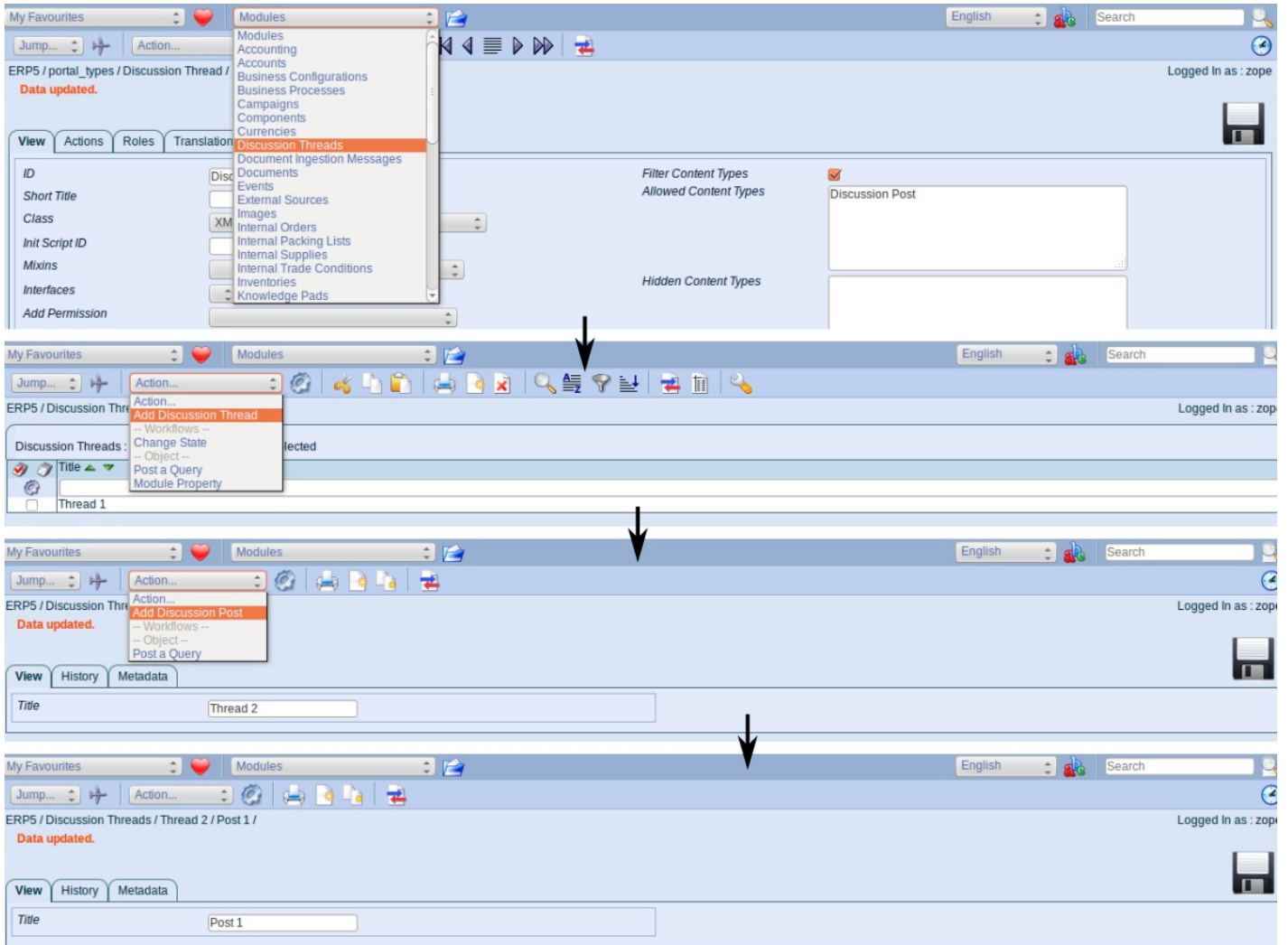
▼ Details

Before creating a Post object, you must verify that "Discussion Post" is a kind of object which can be contained in your Thread-the existed object in your Discussion Threads module.

You must go back on the Discussion_Thread Portal Type in order to check it. The URL should be INSTANCE_ID/portal_types.

In the Allowed Content Type field, you must write "Discussion Post" and save the document. It will then allow your Discussion Thread to contain Posts. Portal types are allowed to contain any other objects, but most of the time content types are filtered.

Create Thread and Post



▼ Details

At this step of the Tutorial, you must have a "Discussion Threads" Module in which you can create "Discussion Thread" documents, in which you can also create "Discussion Post" documents.

Go to **Modules** tab (as shown in the slide) and choose **Discussion Threads**.

After that, click on "Action..." then choose "Add Discussion Thread" (In that module you can see a list of Threads which have been created).

From a Discussion Thread document, click on "Action..." then choose "Add Discussion Post". You then land on a new post, to which you can give a title.

If you go back to the Discussion Thread document, you cannot see the list of Posts which have been created yet. We will now see how to associate a workflow to the Discussion Thread and Posts.