

# Integrate Wonder Into an Existing Application

## Introduction

So you have decided to take the plunge and take advantage of the many useful features of Project Wonder. The most common usage of Project Wonder is to simply integrate the Wonder foundation frameworks of ERJars, ERExtensions and JavaWOExtensions into your project. Doing this fixes many problems with the standard WebObjects frameworks and provides easy access to many configurable features that most typical WebObjects applications need. In this short article, we itemize the absolute minimum to add Wonder to your project and avoid initial integration stumbles.

## Minimum Wonder Integration

- Add ERJars, ERExtensions, WOOgnl and Wonder's JavaWOExtensions frameworks to your project's framework references
- Add ERPrototypes if your EO model uses/will use prototypes
- Remove existing log4j jar if your project had that.
- Make it so your Application class subclasses ERXApplication
- Make it so your Session class subclasses ERXSession
- Make it so your DirectAction classes extend ERXDirectAction
- Make it so your Enterprise Objects extend ERXGenericRecord instead of EOGenericRecord. If you use EOGenerator, simply change your superclass template class declaration and import statements.
- Your Application class' main static method **must** call ERXApplication.main(...) instead of WOApplication.main(...). Your application will fail to initialize and launch otherwise.
- Use ERXEC.newEditingContext() to create new editing contexts in your application
- Add minimal content to Properties file. To get minimal content, create a temporary empty project using the WOrderApplication WOLips project template and copy/paste the contents from that template's Properties file.

Remember



All of these steps are important. Failing to integrate properly may lead to errors during runtime or your application will fail to launch successfully

### Application main method

```
public static void main(String argv[]) {
    ERXApplication.main(argv, Application.class);
}
```

## Easy Extras

If you got the minimum Wonder integration done and your app is working, then adding these extra fundamental useful frameworks are definitely worthwhile from the start:

- Ajax - Just what it says - easy Ajax in your WO apps
  - ERPrototypes - save yourself lots of manhours fiddling with EOModel attributes by using convenient prototypes. This is considered best practice. (Don't forget to select a prototype in your database config in Entity Modeler, otherwise the prototype won't show up when adding/editing an attribute.)
  - ERJavaMail - send all kinds of emails - plain text, html, multi-part, attachments, etc. HTML can be generated from a WOComponent or be set using a HTML String.
  - ERSelenium - functional testing that simulates user interaction with your app's web pages.
- Wonder source is in itself a very useful reference and gives insight into the inner workings of the WebObjects frameworks, so having the source directly linked to your project in your workspace has many advantages. See other tutorials on this wiki that show how to set that up.

For your next project, simply select the WOnderApplication project template instead of the WOApplication template and all the minimum integration tasks itemized above are already done for you. Besides, the WOnder Application project template is so neatly organized that some have commented on how it is a better overall project template for WebObjects projects even if you are not using Wonder.

