

Creating Extensions the Easy Way

By Thad Tilton, ESRI

The X10-DoMatic extension, available for free download, provides an intuitive interface for automatically creating ArcView GIS extensions without Avenue programming. Visit ESRI's ArcScripts Web page (www.esri.com/arcscripts) to download the extension and readme file.

An Overview of ArcView GIS Extensions

ArcView GIS extensions are a great way to deliver new functionality and data to users. Extensions can be added or removed from any ArcView GIS project as needed. Users can benefit from customizations of ArcView GIS that have been developed within the organization without having to learn how to modify a GUI or compile a script.

Extensions are ASCII text files that are similar in format to an ArcView GIS project file (*.apr). An extension file (*.avx), an example of an object database (ODB), contains object definitions. These objects are usually controls such as menus, buttons, or tools. Extensions may also contain ArcView GIS documents, scripts, dialogs, or even entire custom user interfaces.

Objects delivered by an extension are owned by the extension. This means that the original objects loaded into a project from an extension cannot be modified. A user can make edits or modifications to objects in the project, but the original extension objects remain unaltered. This makes extensions a very powerful method for delivering standard ArcView GIS documents and controls to any project.

The Basics of Creating Extensions

An extension begins with a source project. This is simply an ArcView GIS project that contains the controls, documents, or dialogs that the developer wants to deliver to users via the extension. After testing these objects, at least three Avenue scripts—a Make script, an Install script, and an Uninstall script—are needed to produce the extension. These scripts are written automatically by the X10-DoMatic extension.

The Make Script

This script writes the extension to disk. Generally, a Make script will reference all the objects included in the extension, create an Av-

enue extension object, pack all the objects into the extension, define the extension's properties (e.g., name, description, version), write any controlling scripts to the extension, and commit the extension to disk.

The Install Script

This script delivers the objects defined in the extension file (*.avx) to the current project. When an extension is checked in the ArcView GIS Extension dialog box and the OK button is clicked, the Install script executes.

The Uninstall Script

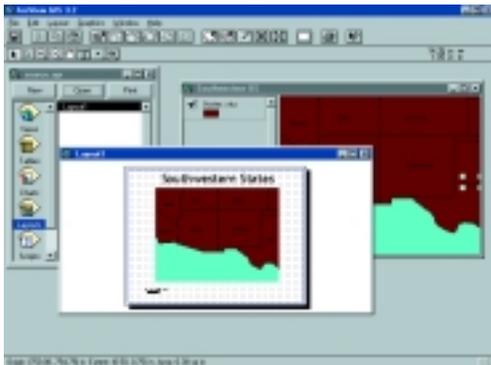
When an extension is unchecked in the Extensions dialog box and the OK button is pressed, the Uninstall script executes and removes all objects delivered by the extension from the current project.

Build an Extension Using the X10-DoMatic Extension

This exercise uses the X10-DoMatic extension to create a sample extension that adds a button to the View GUI and two documents, a view and a layout, to the current project.

Create a Source Project

1. Open a new ArcView GIS project.
2. Create a new view. Click on the Add Theme button and path to the Esridata subdirectory or other location where the sample data that came with ArcView GIS is located. Locate the states.shp shapefile in the USA folder and add it as a new theme.
3. Choose View > Properties, change the view name to United States, set the background color for the view to light blue, and close the View Properties dialog box.
4. Zoom to a favorite part of the country.
5. From the View menu choose Layout. Make a landscape layout of the selected portion of the United States.



Extensions can deliver predefined views and layouts as well as tools and data to users.



The Extension Builder extension is added as an item in the project window.



In the Customize dialog box, create a new button and add an icon, Bex the Dog, to it.



After creating the new button, attach the View.SetName script to it.

Attach the Script to a Button

This sample extension will also include a script called View.SetName that invokes a dialog box for changing the name of a view. This script will be attached to a button in View GUI.

1. Type or copy and paste the code in Figure 1 into a new Script file.
2. Compile it by clicking on the Run button.
3. Choose Script > Properties and rename the script View.SetName.
4. Invoke the Customize dialog box either by returning to the view and double-clicking on any part of the Button bar without an existing button or by going to the project view and selecting Customize from the Project menu.
5. In the Customize dialog box, set the Type to View and the Category to Buttons.
6. The existing buttons for the view document will appear. Hit the New button to create a blank button.
7. Double-click on the icon and pick Bex the Dog as the icon for this button from the available icons.
8. In the properties table below the buttons, double-click on the Click property. Choose the View.SetName script from the Script Manager to attach it to the button.

Test the New Button

Always test controls before writing them to an extension. Make the United States view active and click on the new button (i.e., Bex button).

A dialog box requesting a new name for the view should appear. Typing in a new name should change the text displayed in the view's title bar.

Create a New Extension with X10-DoMatic

From the Project View menu choose File > Extensions and load the X10-DoMatic extension. This extension adds a new type of document to the project called an Extension document. Open a new Extension document by double-clicking on the X10 icon in the project window. In the dialog box asking which GUI to use, choose View.

When the Extension Builder document opens, the button bar and toolbar on the ArcView GIS interface will change depending on which GUI was chosen in the previous step. Choosing <Documents Only> will generate empty button bars and toolbars. Only the custom button is enabled. All standard ArcView GIS View buttons and tools are disabled or grayed-out. Clicking on the custom button invokes a dialog that provides some basic information about the control. Close this dialog after reviewing it.

Add Controls and Documents

The Extension Builder document makes it easy to add custom controls and documents, such as the view and layout documents for this project, to the extension. Clicking the Add Controls button adds all enabled (i.e., custom) controls to the extension being created. Notice that the Scripts list updates and includes the script required for the Bex button. Clicking the Add Documents button on the Extension Builder will bring up the documents that can be added from the source project to the extension. Select the United States view and layout for inclusion in the extension.

Set Extension Properties

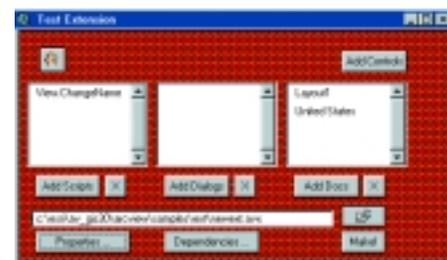
Properties can also easily be added to the extension using the Extension Builder document. Click the Properties button to view or set various properties for the extension. For the sample extension, change the name, description, and version properties as indicated.

- The Name property specifies the name that will appear in the extension's dialog box. Change the name to Test Extension.
- The Description property provides a brief description in the bottom portion of the extension dialog when a user clicks on the extension's name. The description for the Test Extension should read "This is a test of the X10-DoMatic Extension Builder extension. It will deliver a single custom

Continued on page 40

Figure 1

```
theView = av.GetActiveDoc
theView.SetName (MsgBox.Input
("Name:", "New Name", ""))
```



The Extension document makes it easy to add custom controls such as new buttons.



After writing the new extension, Extension Builder will summarize its content in a message box.

Creating Extensions the Easy Way

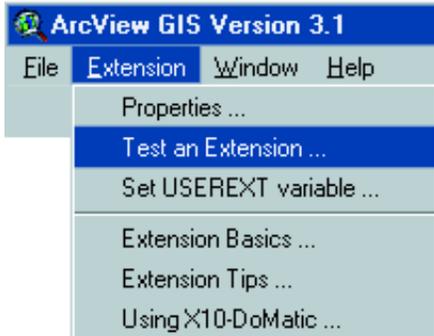
Continued from page 39

button, a view, and a layout document to your project.”

➔ The Version property indicates the extension's current version. Leave the version number at the default value of 1. Click OK in the Properties dialog box to exit.

Specify Output File

Name the output file with .avx extension and indicate where it should be located. Click the button with the open file (folder) icon and browse the drive box list for a location. Click OK after naming and locating the output file.



Before testing the new extension, **close the source project**.

Make the Extension

After creating and testing the objects for the new extension and using the Extension Builder document to add them and to set all applicable extension properties, the final step is to make the extension by writing it to disk. Click the Make button on the Extension Builder document. The extension will be written to the specified file. A message box confirming that the extension was written and summarizing its contents should appear.

Test the Extension

Before you test your extension, **close the source project**. Loading the new extension into the source project will corrupt the project.

1. Save and close the source project.
2. Open a new project to test the extension.
3. In the new project, double-click on the X10 icon to create a new Extension Builder document.
4. Choose <Documents Only> for the GUI type.
Note: Though this test will not create an extension this time, a new document is required to access menu choices from the Extension Builder interface.
5. With the Extension document active, choose Extension > Test an Extension. A dialog warning you not to load an extension into its source project will appear. Identify the extension to test so that the \$USEREXT environment variable will reset to point to the directory containing the new extension.

6. Choose File > Extensions and locate the new extension, Test Extension, in the dialog box. Click on the name and notice that the extension description appears in the lower portion of the dialog. Check the box next to the extension and press OK to load it into the test project.
7. Check to see if the view and layout documents and Bex button loaded properly. Open the view and layout documents delivered by the extension to see if they look as they did in the source project. Locate and click the Bex button to see if it works properly.

8. Return to the Project View and choose File > Extensions. Uncheck Test Extension in the dialog to unload the extension. Message boxes will ask if you want to remove the documents from the project. Click Yes to remove the view and layout documents.

More information on the Extension Builder extension may be found in the x10 readme file. Thad Tilton, author of the X10-DoMatic extension, can be contacted via E-mail at tilton@esri.com. 