# 1

Introducing ArcGIS Desktop development

The ESRI® ArcGIS® Desktop Developer Guide is an introduction for anyone who wants to customize or extend ArcGIS Desktop applications, such as ArcMap™ or ArcCatalog™.

This chapter introduces the ArcGIS Desktop development environment in the context of the ArcGIS product family.

Topics covered in this chapter include an ArcGIS 9 developer overview, an ArcGIS Desktop developer overview, using this guide, and ArcGIS developer resources.

ArcGIS 9 is an integrated family of geographic information system (GIS) software products for building a complete GIS. It is based on a common library of shared GIS software components called ArcObjects™. ArcGIS 9 consists of four key parts:

- **Desktop GIS**, an integrated suite of advanced GIS applications.

- **Embedded GIS**, embeddable GIS component libraries for building custom applications using C++, COM, .NET, and Java™.

- **Server GIS**, a shared library of GIS software objects used to build server-side GIS applications in enterprise and Web computing frameworks. Used for building both SOAP-based Web services and Web applications using .NET/ASP and Java/JSP.

- **Mobile GIS**, GIS Web services to publish maps, data, and metadata through open Internet protocols.
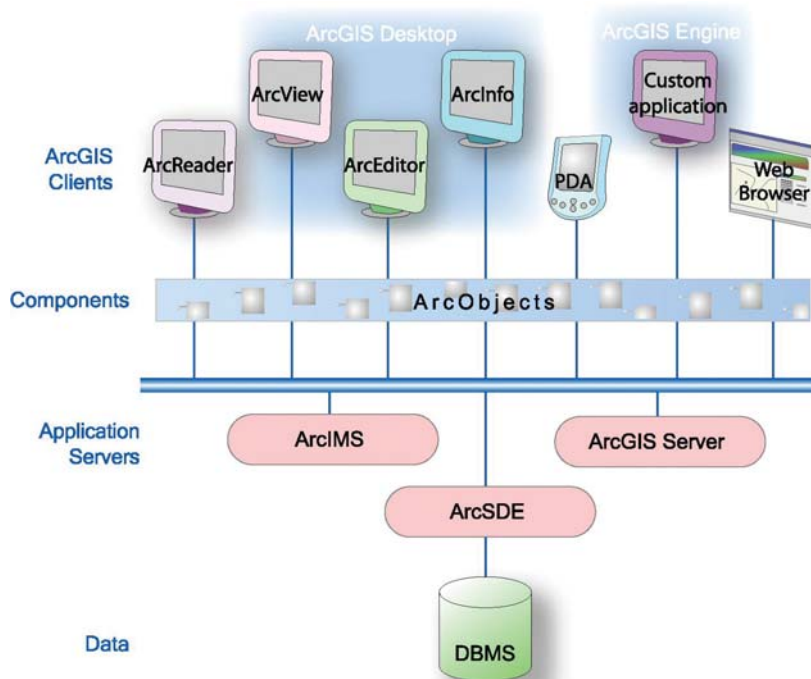
ArcSDE® Gateway is an interface for managing geodatabases in numerous relational database management systems (RDBMSs).

ArcGIS provides a scalable framework for implementing GIS for a single user or for many users on desktops and servers. This book will be of greatest use to developers who want to use the ArcGIS Desktop Developer Kit to customize and extend the ArcView®, ArcEditor™, or ArcInfo® desktop products. This book provides a general explanation of the options and opportunities available to developers with ArcGIS Desktop. Several scenarios will illustrate with code examples the different types of customization that can be developed with the ArcGIS Desktop Developer Kit.

ArcGIS is a platform for building geographic information systems. ArcGIS 9 will extend the system with major new capabilities in the areas of geoprocessing, 3D visualization, and developer tools. Two new products, ArcGIS Engine and ArcGIS Server, are introduced at this release, making ArcGIS a complete system for application and server development.

There is a wide range of possibilities when developing with ArcGIS, including the following:

- Configure/Customize ArcGIS applications, such as ArcMap and ArcCatalog.

- Extend the ArcGIS architecture and data model.

- Embed maps and GIS functionality in other applications with ArcGIS Engine.

- Build and deploy custom desktop applications with ArcGIS Engine.

- Build Web services and applications with ArcGIS Server.

ArcGIS 9 has a common developer experience across all ArcGIS products (Engine, Server, and Desktop). This book focuses on customizing and extending the ArcGIS Desktop applications. Developers wanting to build custom standalone applications or work with ArcGIS Server should refer to the *ArcGIS Engine Developer Guide* and the *ArcGIS Server Administration and Development Guide*, respectively.
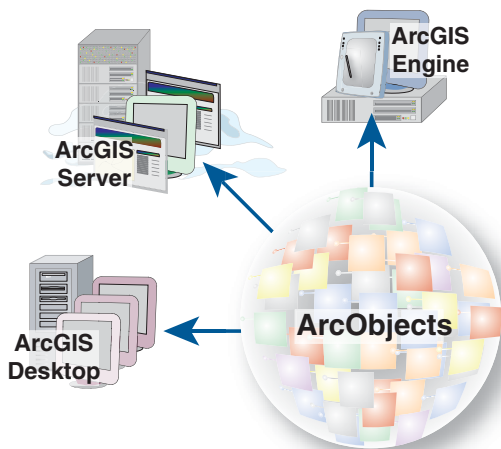

The ArcGIS system is built and extended using ArcObjects software components. ArcObjects includes a wide variety of programmable components, ranging from fine-grained objects (for example, individual geometry objects) to coarse-grained objects (for example, a map object to interact with existing ArcMap documents), which aggregate comprehensive GIS functionality for developers.

Developers work with ArcObjects using standard programming frameworks to extend ArcGIS Desktop, build custom applications with ArcGIS Engine, and implement server GIS applications for various enterprise purposes using ArcGIS Server.

Each of the ArcGIS product architectures built with ArcObjects represents alternative application development containers for GIS software developers, including desktops, embeddable engines, and servers.

ARCGIS SOFTWARE DEVELOPER KITS

The ArcGIS Engine, Server, and Desktop have software developer kits (SDK) for working with ArcObjects. The developer kits provide everything needed to be successful with ArcObjects, including developer documentation, samples, and object model diagrams.
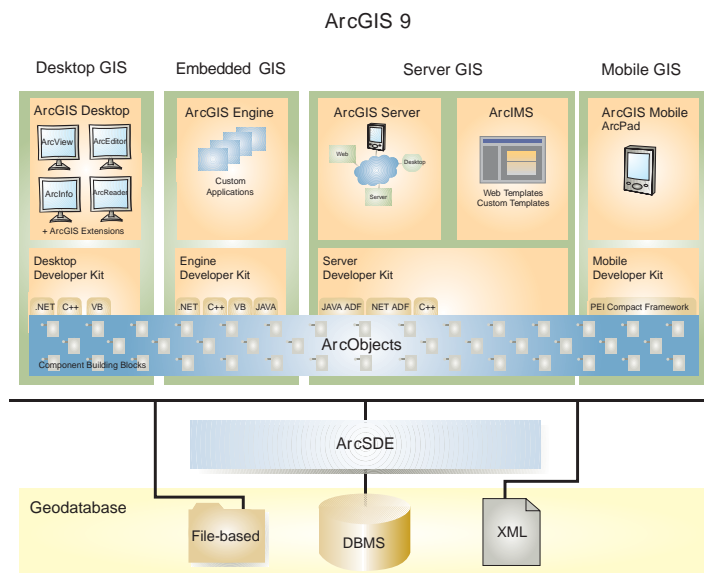
ArcGIS 9

The ArcGIS Desktop Developer Kit

ArcGIS Desktop includes a series of Windows® desktop application frameworks—ArcMap, ArcCatalog, ArcToolbox™—and more, each with user interface (UI) components. ArcGIS Desktop is available at three functional levels (ArcView, ArcEditor, and ArcInfo) and can be customized and extended using the ArcGIS Desktop Developer Kit.

The software developer kit for ArcGIS Desktop is included with ArcView, ArcEditor, and ArcInfo and supports the COM and .NET programming frameworks. Many users apply the ArcGIS Desktop Developer Kit to add extended functions, new GIS tools, custom user interfaces, and full extensions for improving professional GIS productivity of ArcGIS Desktop.

The ArcGIS Engine Developer Kit

ArcGIS Engine is a simple, application-neutral programming environment for ArcObjects. The ArcGIS Engine Developer Kit provides a series of embeddable ArcGIS components that are used outside the ArcGIS Desktop application

framework (for example, mapping objects are managed as part of ArcGIS Engine, rather than in ArcMap). Using the ArcGIS Engine Developer Kit, developers build focused GIS solutions with simple interfaces to access any set of GIS functions or can embed GIS logic in existing user applications to deploy GIS to broad groups of users. ArcGIS Engine has a COM, .NET, Java, and C++ application programming interface (API) for developers.

The ArcGIS Server Developer Kit

ArcGIS Server defines and implements a set of standard GIS Web services (for example, mapping, data access, and geocoding), as well as supports enterprise-level application development based on ArcObjects for the server.

The ArcGIS Server Developer Kit enables developers to build central GIS servers to host GIS functions that are accessed by many users, perform back office processing on large central GIS databases, build and deliver GIS Web applications, and perform distributed GIS computing.

ArcGIS Desktop includes a suite of integrated applications, including ArcMap, ArcCatalog, and ArcToolbox. By using these applications and interfaces in unison, you can perform any GIS task, simple to advanced, including mapping, geographic analysis, data editing and compilation, data management, visualization, and geoprocessing.

WHAT ARE ArcView, ArcEditor, AND ArcInfo?

ArcGIS Desktop is the information authoring and usage tool for GIS professionals. It is scalable as three separate software products to meet the needs of many types of users.

ArcView provides comprehensive mapping and analysis tools along with simple editing and geoprocessing.
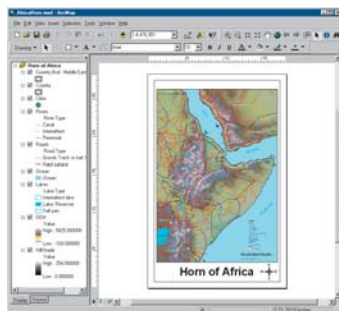
ArcEditor includes advanced editing capabilities for shapefiles and geodatabases in addition to the full functionality of ArcView.

ArcInfo is the flagship ArcGIS Desktop product. It builds on the functionality of ArcEditor with advanced geoprocessing. It also includes the legacy applications for ArcInfo Workstation.

Because ArcView, ArcEditor, and ArcInfo all share a common architecture, users working with any of these GIS desktops can share their work with other users. Maps, data, symbology, map layers, custom tools and interfaces, reports, metadata, and so on, can be accessed interchangeably in all three products. This means that you benefit from using a single architecture, minimizing the need to learn and deploy several different architectures.

New capabilities can be added to all seats through a series of ArcGIS Desktop extensions from ESRI and other organizations. Users can also develop their own custom extensions to ArcGIS Desktop by working with ArcObjects, the ArcGIS software component library. Users develop extensions and custom tools using standard Windows programming interfaces, such as COM and .NET.

ArcObjects is a framework that lets you create domain-specific components from other components. The ArcObjects components collaborate to serve every data management and map presentation function common to most GIS applications. ArcObjects provides an infrastructure for application customization that lets you concentrate on serving the specific needs of your clients.

ArcMap

ArcMap is the central application in ArcGIS Desktop. It is the GIS application used for all map-based tasks, including cartography, map analysis, and editing. In this application, you work with maps. Maps have a page layout containing a geographic window, or a data frame, with a series of layers, legends, scalebars, North arrows, and other elements. ArcMap offers different ways to view a map's geographic data and layout views in which you can perform a broad range of advanced GIS tasks.

ArcMap is used for mapping and editing tasks as well as map-based analysis.
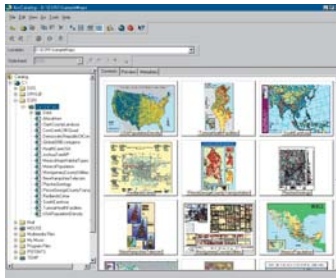
ArcCatalog is used for managing your spatial data holdings, defining your geographic data schemas, and recording and viewing metadata.

ArcCatalog

The ArcCatalog application helps you organize and manage all of your GIS information (maps, globes, datasets, models, metadata, services, and so on). It includes tools to:

- Browse and find geographic information.

- Record, view, and manage metadata.

- Define geodatabase schemas and designs.

- Administer an ArcGIS Server.

- Search for and discover GIS data on local networks and the Web.

GIS users apply ArcCatalog to organize, find, and use GIS data as well as to document their data holdings using standards-based metadata. A GIS database administrator uses ArcCatalog to define and build geodatabases. A GIS server administrator uses ArcCatalog to administer the GIS server framework.

Customizing ArcGIS Desktop applications

You may want the ArcMap and ArcCatalog interfaces to reflect your own preferences and the way you work.

You can customize ArcMap and ArcCatalog in many ways. Here are some examples:

- Position toolbars in a specific area of the application.

- Group commands in a way that works best for you.

- Add new macros or load custom commands from another source.

- Always work with the same geographic data (via templates).
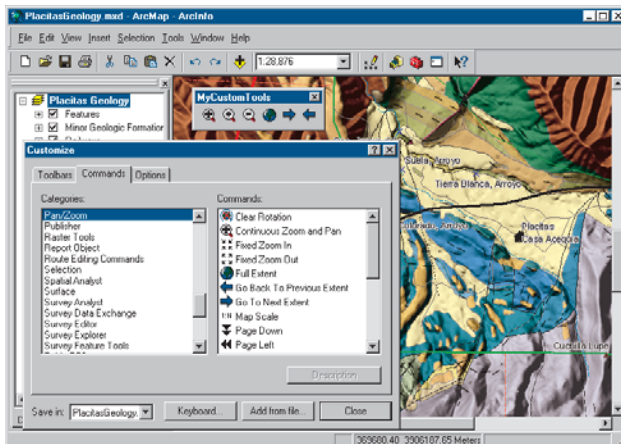
The Customize dialog box is used to create new toolbars and add or remove controls.

If you work in a larger organization, others may want you to develop a customized work environment for them. You can handle many customization tasks without writing a single line of code. In fact, you may be able to instruct others on how to use the customization environment to create the look and feel they want on their own. You can change or create toolbars, menus, shortcut keys, and so on, to help you do your work in the most efficient way. You can provide additional functionality by linking code you or others have written to menu commands or tools.

Several toolbars are provided with ArcMap and ArcCatalog, but you may want to create new toolbars to organize commands that you often use together or to contain buttons that run your custom scripts.
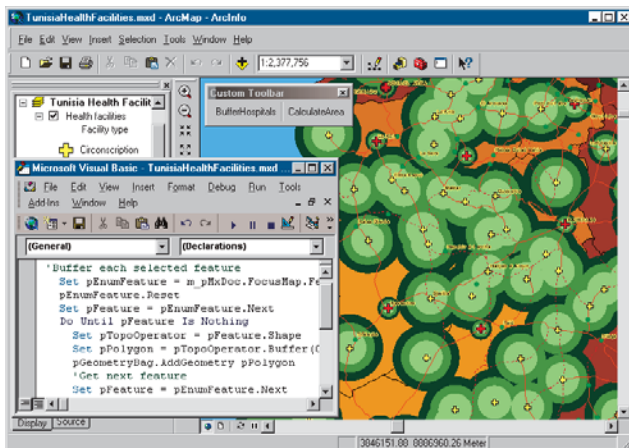
Writing VBA macros in ArcGIS applications

Both ArcMap and ArcCatalog come with Visual Basic® for Applications (VBA). VBA is not a standalone program; it's embedded in the applications. It provides an integrated programming environment, the Visual Basic Editor (VBE), which lets you write a Visual Basic (VB) macro, then debug and test it right away in ArcMap or ArcCatalog. A macro can integrate some or all of VB's functionality, such as using message boxes for input, with the extensive object library that ArcMap and ArcCatalog expose. The ESRI object libraries are always available to you in the VBA environment.

There is an easy way to create custom commands with VBA. You can create a new button, tool, combo box, or edit box (collectively called UIControls), then attach code to the control's events, such as what happens when you click a button. After you have created it, you can drag this new control onto a toolbar.

As mentioned, you should start development by using the VBA environment in one of the existing ArcGIS applications. VBA is a simple programming language with many utilities, such as design time code completion and the Object Browser that will help you assemble code quickly.

Here are more reasons to choose the VBA environment:

A buffer command created in VBA

• It's fast and easy to create, test, and debug macros inside ArcMap and ArcCatalog.

• The standard ESRI type libraries are already referenced for you.

• Important global variables, such as the Application and Document, are available.

• It's simple to assemble UI forms using VBA and ActiveX® components.

• It's straightforward to integrate VBA code with new ArcObjects UIControls.

• It's relatively easy to migrate VBA code to VB ActiveX Dynamic Link Library (DLL) projects.

• Many code samples available in the help system are macros that can be cut, pasted, and run within the VBA environment.

Writing custom components to extend the ArcGIS Desktop applications

You don't have to use VBA to create custom commands and toolbars—in fact, in some cases, your custom commands and toolbars may require you to use another development environment. You can create custom objects in any programming language that supports the Microsoft® Component Object Model. Custom commands or toolbars created outside VBA are often distributed as ActiveX DLLs. If you have created some custom commands and toolbars, or someone else has given you an ActiveX DLL containing custom commands and toolbars, you can easily add these objects to ArcMap or ArcCatalog. After adding a custom object to ArcMap or ArcCatalog, you can use it as you would any built-in command or toolbar.

If you want to use a programming language other than VBA, or if you want to package ArcObjects functionality into a COM DLL, EXE, OCX, or .NET assembly, you will have to work outside the VBA development environment. This approach generally requires creating a project, referencing the ArcObjects type libraries required, implementing ArcObjects interfaces, adding code, then compiling the source into a binary file.

Following are some advantages of building custom components:

• They can be easily delivered to end users via custom setup programs.

• You can hide your code and logic in a binary file.

• You can extend and customize virtually every aspect of the ArcGIS technology.

Components can be broadly categorized into two areas of customization: those that reside at the application level, such as custom buttons, toolbars, windows, and extensions, and those that reside at the geodatabase level, such as custom feature class extensions and features. Some of these more advanced customizations cannot be accomplished through the VBA environment.

The Production Line Tool Set (PLTS) is an example of a complex application extension created with ArcObjects.
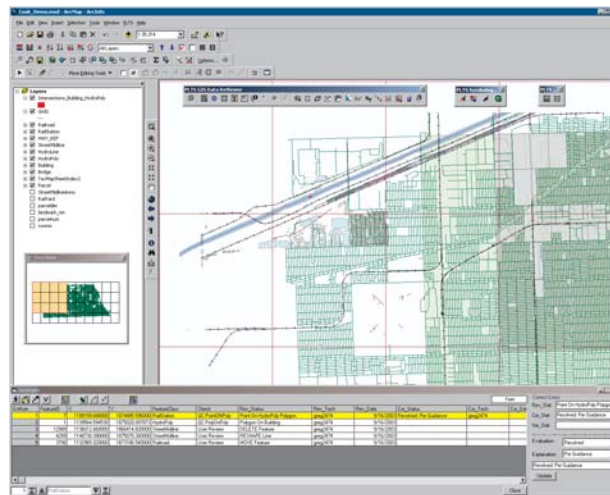
The *ArcGIS Desktop Developer Guide* is an introduction for anyone who wants to customize or extend the ArcGIS Desktop applications, such as ArcMap or ArcCatalog.

This guide will help you become a desktop developer by walking you through numerous VBA code samples and providing a problem-solving guide and developer scenarios. Although the samples documented in this guide may not solve your immediate problem, they will serve as a framework or template on which you can build a more specific or complex solution.

To serve the greatest base of developers, most of the code samples in this guide are written in VBA. As necessary, some code samples are written in Microsoft Visual Basic, VB .NET, or Visual C++®.

CHAPTER GUIDE

Chapter 1, 'Introducing ArcGIS Desktop development', gives you an overview of the ArcGIS 9 product family, the desktop developer framework, the types of customizations that can be made, and additional resources.

Chapter 2, 'ArcGIS software architecture', describes the underlying technology used to create the ArcGIS Desktop products, ArcObjects, and the software architecture common to the ArcGIS product family.

Chapter 3, 'Developing for ArcGIS Desktop applications', guides you through the basics of ArcGIS Desktop development. The chapter begins with some desktop application framework theory, then walks you through some small tutorials for toolbar customization and VBA macro writing. The chapter concludes with the theory and steps in creating custom components that extend the ArcGIS Desktop framework.

Chapter 4, 'Licensing and deployment', discusses license considerations when developing your customizations and how to package and deploy your customizations, including VBA macros and components, to other users.

Chapter 5, 'Developer scenarios', guides you through the creation of several types of components that can plug into the ArcGIS Desktop applications.

Appendix A, 'Developer environments', describes in more detail the Microsoft Component Object Model, the foundation for ArcObjects, and continues with a language reference guide for each API supported by ArcGIS. These guides discuss issues and considerations while developing ArcGIS with a particular API.

Appendix B, 'Reading the object model diagrams', discusses how to read and interpret the ArcGIS object model diagrams. Understanding the structure and relationships of ArcObjects represented in these diagrams is essential in ArcGIS development.

Appendix C, 'Illustrated code samples', shows numerous VBA code snippets for some common desktop applications. These are illustrated with components from the ArcGIS object model to help you understand the structure and relationships.

Appendix D, 'ArcObjects problem-solving guide,' presents a methodology to help you solve real-world ArcObjects programming tasks that customize or extend the ArcGIS Desktop applications.

Appendix E, 'UIControls', shows class and interface diagrams for UIControls, which are VBA-based commands with interfaces available only in VBA.

Appendix F, 'Bibliography', is not intended as a complete resource, but it does contain many of the everyday references that ESRI developers use when developing Visual Basic, Visual C++, Visual Studio .NET code, and ArcObjects.
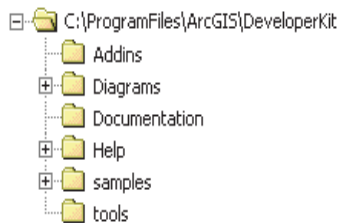
The following topics describe some of the additional resources available to you as a developer. These include books and guides and various help systems.

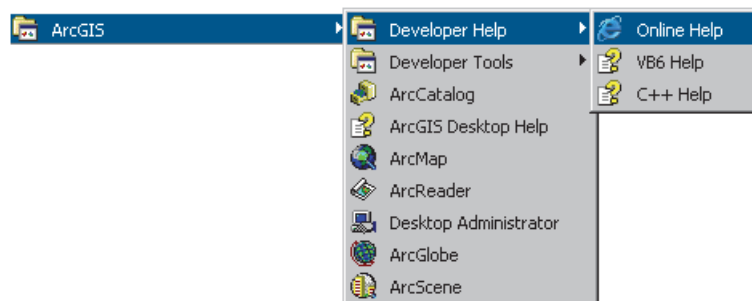### ARCGIS SOFTWARE DEVELOPER KIT

The ArcGIS SDK is a collection of diagrams, utilities, add-ins, samples, and documentation geared to help developers implement custom ArcGIS functionality.

ArcGIS Developer Help system

The ArcGIS Developer Help system is the gateway to all SDK documentation, including help for the add-ins, developer tools, and samples; in addition, it serves as the complete syntactical reference for all object libraries.

Each supported API has a version of the help system that works in concert with it. Regardless of the API you choose to use, you will see the appropriate library reference syntax and have a help system that is integrated with your development environment. For example, if you are a Visual Basic 6 developer, you will use ArcGISDevHelp.chm, which has the VB6 syntax and integrates with the VB6 integrated development environment (IDE), thereby providing F1 help support in the Code window.

The help systems reside in the DeveloperKit\Help folder but are typically launched from the Start menu or F1 Help in Visual Basic 6 and Visual Studio .NET 2003. The graphic below shows the Start menu options for opening the help systems.
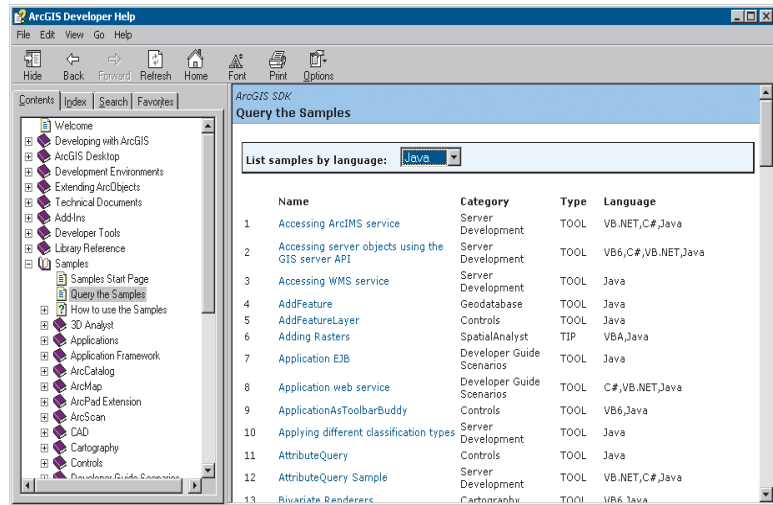

Samples

The ArcGIS developer kit contains more than 600 samples, many of which are written in several languages. The samples are described in the help system, and source code and project files are installed in the DeveloperKit\samples folder. The help system's table of contents for the samples section mirrors the samples directory structure.

The help system organizes samples by functionality. For example, all the geodatabase samples are grouped under Samples\Geodatabase. Most first-tier groupings are further subdivided. You can also find samples in the SDK using the 'Query the Samples' topic in the help system, which lists all the samples alphabetically; in addition, you can sort the list by language. For example, you can elect to only list the available VB6 samples.
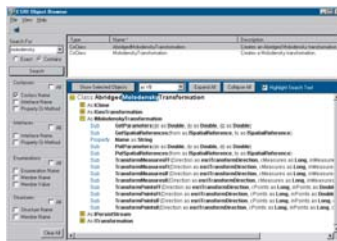

A typical SDK installation

Installing the sample source code and project files is an option in the Desktop Developer Kit install. The samples are installed under the ArcGIS\DeveloperKit\samples folder. If you don't have this folder on your computer, you can rerun the install program and check Samples under Developer Kit.
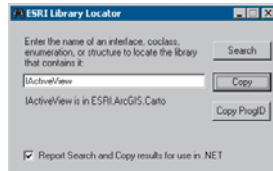
ESRI Library Locator

ESRI Object Browser

Developer tools

The ArcGIS developer tools are executables that ESRI has provided to facilitate your ArcObjects development. For example, if you are a Visual Basic 6 desktop developer, you will likely use the Categories.exe tool to register components in component categories.

The list below features some of the more important developer tools available with ArcGIS Desktop. Refer to the help system for more developer tool details and instructions.

• Component Categories Manager—Registers components within a specific component category

• Library Locator—Identifies an object library containing a specified interface, coclass, enumeration, or structure

• ESRI Object Browser—Lets you explore the structure of ArcObjects, over-coming certain limitations of other standard object browsers

The developer tools are installed in the DeveloperKit\tools folder. There is one exception—the Component Category Manager is located in the ArcGIS\bin folder.

Add-ins

The ESRI add-ins automate some of the tasks performed by the software engineer when developing with ArcObjects, as well as provide tools that make debugging code easier. ESRI provides add-ins for the Visual Basic 6 IDE and the Visual Studio .NET IDE. Listed below are the add-ins available for these development environments.

Visual Basic 6

- ESRI Align Controls With Tab Index—Ensures control creation order matches tab index

- ESRI Automatic References—Automatically adds ArcGIS library references

- ESRI Code Converter—Converts projects from ArcGIS 8.x to ArcGIS 9.x

- ESRI Command Creation Wizard—Facilitates the creation of commands and tools

- ESRI Compile and Register—Aids in compiling components and registering these in desired component categories

- ESRI ErrorHandler Generator—Automates the generation of error handling code

- ESRI ErrorHandler Remover—Removes the error handlers from the source files

- ESRI Interface Implementer—Automatically stubs out implemented interfaces
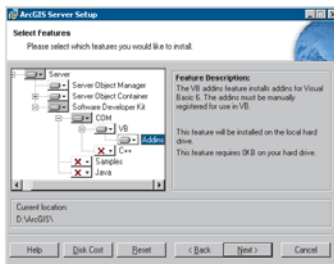
Visual Studio .NET

- ESRI Component Category Registrar—Stubs out registration functions to enable self-component category registration

- ESRI .NET Code Converter—Converts projects from ArcGIS 8.x to ArcGIS 9.x

- ESRI GUID Generator—Inserts a Global Unique Identifier (GUID) attribute

The .NET add-ins are automatically installed during setup if a version of Visual Studio .NET 2003 is detected; the Visual Basic 6 add-ins are only installed if you select them on the install. Once installed the add-ins have to be registered manually using the .bat files in the Addins directory.

Visual Basic 6 add-ins are only installed if you select them on the install.

THE ARCGIS DEVELOPER DOCUMENTATION SERIES

This guide is part of the ArcGIS Developer documentation series.

The *ArcGIS Engine Developer Guide* provides information for developers who want to create applications based on ArcGIS Engine. ArcGIS Engine allows you to embed GIS functionality within other applications and create desktop-like applications using the supplied ArcGIS controls, such as MapControl, Toolbar, and PageLayout controls. ArcGIS Engine is also based on ArcObjects components and may be programmed through a number of APIs.

The *ArcGIS Server Administrator and Developer Guide* is for developers who will create ArcGIS Server applications and customizations. At the core of ArcGIS Server is a rich ArcObjects object library that can be consumed in Web applications and Web services to deliver advanced GIS functionality to a wide range of users who interact with the server through Web browsers and other thin client applications. ArcGIS Server may also be programmed through a number of APIs.

ARCGIS DEVELOPER ONLINE WEB SITE

ArcGIS Developer Online is the place to find the most up-to-date ArcGIS 9 developer information, including sample code, technical documents, object model diagrams, and the complete object library reference.

The Web site is a reflection of the ArcGIS Developer Help system, except it is online and, therefore, more current. The Web site has some additional features including an advanced search utility that enables you to control the scope of your searches. For example, you can create a search that only scans the library reference portion of the help system.

Visit the site at *http://arcgisdeveloperonline.esri.com*.
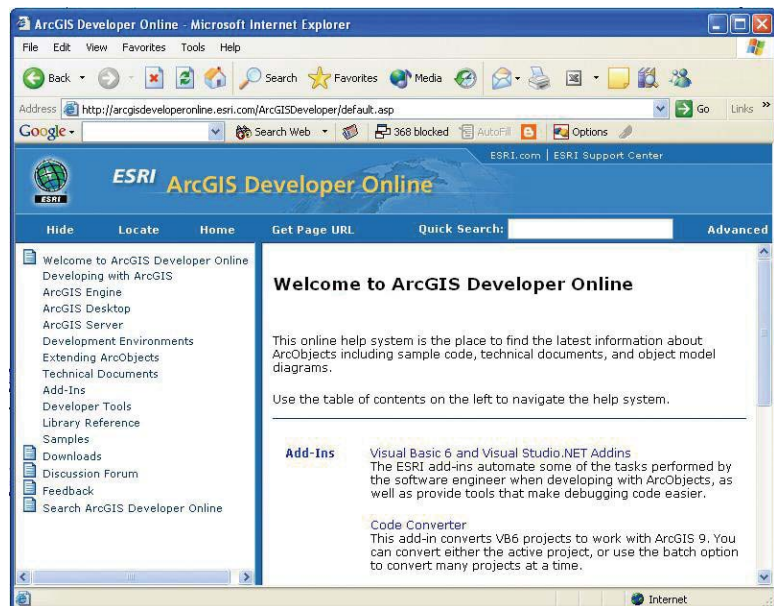
ESRI SUPPORT CENTER

The ESRI Support Center at *http://support.esri.com* contains software information, technical documents, samples, forums, and a knowledge base for all ArcGIS products.

ArcGIS developers can take advantage of the forums, knowledge base, and samples sections in particular to aid in development of their ArcGIS applications.

TRAINING

ESRI offers a number of instructor-led and Web-based training courses for the ArcGIS Desktop developer. These courses range from the introductory level for VBA to more advanced courses in component development with specific APIs.

For more information visit *http://www.esri.com* and click the Training and Events tab.

The ESRI Virtual Campus can also be found directly at *http://campus.esri.com/*.