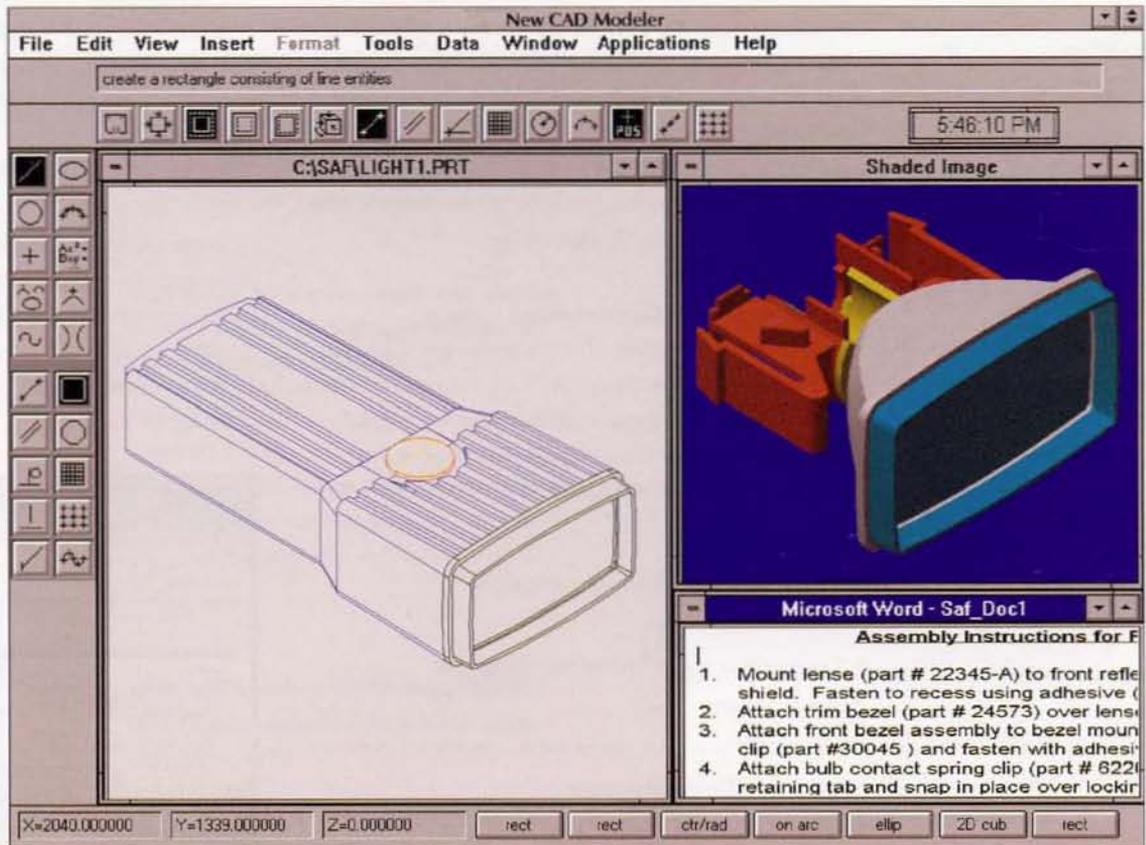




CADKEY® OBJECT Developer



right: Sample Application
created with the CADKEY
OBJECT Developer.

The CADKEY® OBJECT Developer (CODE™) is a self-contained, software development toolkit for 32-bit applications written for use with the Windows™ and Windows NT™ operating systems. CODE is a collection of C++ objects and modules, an Application Framework (AFX), and libraries to support ObjectStore™, the leading client/server database developed by Object Design, Inc. CODE is the premier toolkit for developers who want to cross the threshold into object-oriented programming for Windows.

CODE plugs directly into the Microsoft® Foundation Class (MFC) and gives you cross-platform compatibility and allows you to use off-the-shelf tools, like Microsoft Visual C++ to develop your own graphical user interface (GUI) and augment your application. The direct link with MFC and Object Linking and Embedding (OLE/2) architecture gives you a tightly integrated development solution for Windows. This integration makes it easy to produce OLE container/server products that require high-end graphics, modeling, and object-oriented database capabilities.

If you develop for design, multimedia, graphics, or database markets, or if you require high-end graphic and database capabilities for standalone or AFX applications, CODE allows you to broaden your market and develop products for Windows faster and easier.

OBJECT Developer – the ultimate

Applications:

Manufacturing
Building
Modeling
Database
Graphics
Visualization
Multimedia

CADKEY OBJECT Developer is the ultimate Microsoft Windows and Windows NT toolkit that provides a flexible framework, suitable for a variety of graphical and non-graphical applications. CODE is written entirely in C++ and makes full use of object-oriented programming techniques. An extensive collection of class libraries and the ability to extend through Dynamic Link Libraries (DLL), help you shorten and simplify your software design cycle.

The Applications Development Core

The CODE development core is made up of several application programming interface (API) subsystems. These subsystems support the creation of standalone and AFX programs which may also serve as OLE container/server applications. Components of the main API are:

- Object-oriented database manager
- Graphics pipeline
- Geometric/solids modeling subsystem
- Math library
- Object libraries for parametric/instanced geometry
- Data translator subsystem
- ObjectStore libraries
- Enhanced User Interface Management Subsystem (UIMS) and utility subsystems

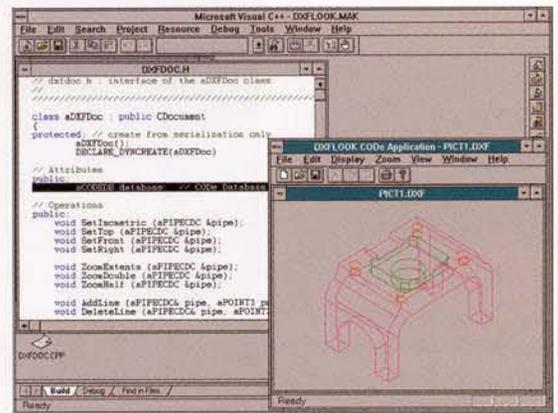
The Applications Development System

Much of CADKEY OBJECT Developer is derived from MFC to simplify development of your application. CODE provides class extensions to the graphics subsystem, process management, application control, document management component, and control validation that enhance the capabilities of MFC. These extensions to MFC are reusable, extensible, scalable, and modular making CODE a highly flexible and modular environment for developing a host of applications.

The CODE UIMS is derived from MFC and has been enhanced to provide specialized functions such as

creating multiple viewports within a single document. CODE contains other class library enhancements for dynamic menu loading, dynamic toolbars, extended string, and file I/O handling. You can easily configure and translate your program interface into foreign languages by using text-based menus (.mnu files).

In addition, the link to MFC makes it easier to develop your programs by allowing you to take advantage of off-the-shelf tools. With tools such as AppWizard, you can create the skeleton and drop in the necessary code, subsystems, and classes for your application.



Building a DFX viewer application using Visual Workbench, MFC and CODE.

State-of-the-art, object-oriented, client/server database utilizing ObjectStore

CADKEY OBJECT Developer includes access to ObjectStore, a leading client/server, object-oriented database through a specialized class library. ObjectStore provides flexibility and is well suited to a variety of applications from multimedia to information-based programs. ObjectStore also provides transaction processing (to ensure against system failure), version control, security, and has cross-application compatibility with other applications using ObjectStore.

ate Microsoft® Windows™ 3.1 & NT development

Requirements:

Operating system:

Microsoft® Windows NT™
Version 3.5

Compiler:

Microsoft Visual C++
32-bit, Version 2.0/
MFC Version 3.0

Graphics Pipeline Connects Directly to MFC

A specialized bi-directional graphics pipeline provides seamless support for graphics display and interaction, printing, and plotting. This pipeline connects directly to MFC windows and enables you to obtain user interface and display data through this or other Graphics Display Interface (GDI) devices supported by MFC.

CODE improves development of graphical applications by providing full viewing control for rotation, scaling, translation, and user-interactive selection. The pipeline includes powerful, CAD-like features enabling you to create perspective, hidden-line removal and shaded images.

Data Translation Subsystem for AutoCAD®, DataCAD®, and CADKEY files

CADKEY OBJECT Developer includes full support for bi-directional translation of AutoCAD DFX® and DWG® files, DataCAD, and CADKEY drawing files, as well as a native, metafile format. This capability allows AFX and standalone application developers to import and export these files into their products.

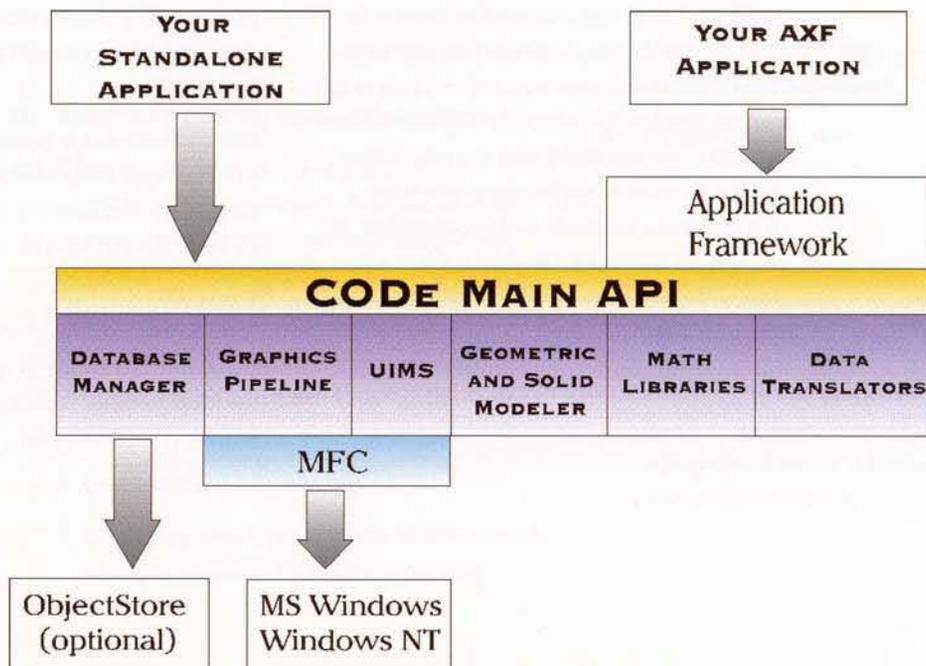
Powerful Geometric Modeling Subsystem with Built-in Solids

This subsystem provides a 2D/3D wireframe and polyhedral solid modeling engine with a complement of object generators. CODE offers complete editing control for 2D/3D curve trimming and solids splitting, punching, chamfering, and boolean operations. Attribute control, analysis, and verification tools give you added functionality for modeling and other graphical applications.

Object libraries for parametric and instanced geometry

CADKEY OBJECT Developer provides complete support for instanced geometry which facilitates multiple occurrences of the same object. CODE also supports used-defined instanced geometric objects which may be parametric, variational, or table-driven. This capability provides a flexible method for updating information.

right: CADKEY OBJECT Developer system structure



nt toolkit.

Hardware:

Windows NT compatible PC (16MB minimum), 24 MB recommended, VGA or better graphics, (256 colors required for rendering applications), CD ROM drive

Extensive Math Libraries

The math libraries support an extensive collection of math operations, including:

- 3x3 and 4x4 matrix operations for viewing and mapping
- 2D and 3D intersection and tangency calculations for points, curves, and surfaces
- Mass property solvers
- A wide assortment of analytic geometry functions

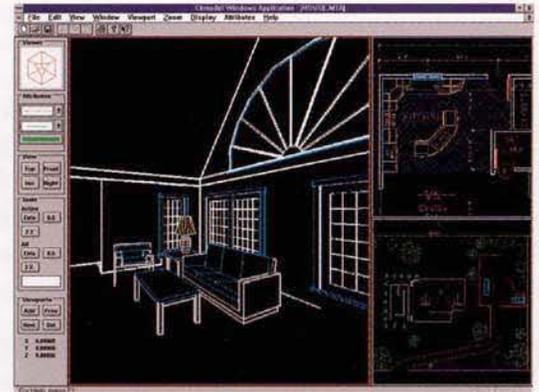
You may use one of two basic methods when writing an application using CADKEY OBJECT Developer; standalone and AFX development.

Standalone Application Development

In a standalone application, you directly access components through the API. This method gives you the ultimate freedom in designing a user interface and database structure. Also, the architecture of CODE eliminates the need to *bundle* your application with another program (such as a CAD system) to produce a complex, turnkey graphics or database product.



Windows sample interface. Picture created by Sweet's Electronic Publishing.



Sample AFX application developed using CODE.

AFX Developed Applications

The application framework for CODE is derived from the MFC AFX making it compatible with the application architecture. The CODE AFX provides a layer on top of the development core that includes process control, data capture, state management, and multi-language support. The AFX is a complete application and screen layout that you can build your application on. Since the AFX is modular, you have the flexibility to *plug/unplug* features to suit your needs.

What does CADKEY OBJECT Developer include?

The CADKEY OBJECT Developer ships on a CD-ROM that includes:

- Object libraries, source files, development tools, MAKE files and scripts
- An extensive user guide and MS-HELP based reference manual
- Sample Application Framework
- Source code and executables for sample applications that you can test and run. You can use these samples as part of your application or as reference material.
- CADKEY OBJECT Developer overview

OBJECT Developer

Questions and Answers

Q: What are the system requirements for CODE?

A: You will need Windows NT 3.5, 16MB of RAM minimum (24MB recommended), VGA or higher, and Microsoft 32-bit Visual C++ Version 2.0/MFC Version 3.0.

Q: Do I need to incorporate the entire CADKEY OBJECT Developer library to build simple applications?

A: No. Since CODE is built entirely using object-oriented design concepts, you can unplug modules at several structural levels without compromising the functionality provided by low level libraries.

Q: Will one version of Cadkey Object Developer support Windows and Windows NT?

A: Yes, however Windows support is through the WIN32S API and Windows NT is required to build 32-bit Windows applications.

Q: What do I gain by using ObjectStore?

A: ObjectStore is fast becoming an industry standard in object-oriented database management systems. It is an extremely flexible and powerful tool for applications where you need to manage large, complex databases efficiently. Some key features provided by ObjectStore are:

- Client/server architecture with full network support
- Cross-database referencing
- Versioning which provides the ability to track multiple versions of a database schema.

Q: Do I have to use ObjectStore?

A: No. You can develop non-ObjectStore products by using the Cadkey Object Developer binary metafile as the storage medium for your data.

Q: How can I link my applications with Excel, Word, or other Windows-based products?

A: Data can be transferred through the clipboard or linked using Object Linking and Embedding (OLE/2).

Q: Should I use this toolkit instead of AutoCAD or some other CAD-oriented product?

A: Yes. You can create sophisticated, standalone applications without requiring another product in order to run. CODE is fully object-oriented, scalable, and modular whereas AutoCAD and other monolithic systems are not.

Q: Do you support any photo-realistic rendering packages?

A: Cadkey Object Developer could support any rendering package that works with MFC.

Q: Can I redistribute products created using the Cadkey Object Developer?

A: Yes, however the product you receive is a single-user version only. You may contact Cadkey for a Run-time License Agreement and full description packet outlining redistribution fees.



CADKEY

Questions and Answers

Cadkey, Inc.
4 Griffin Road North
Windsor, Connecticut
06095-1511

Phone: 203-298-8888
Fax: 203-298-6484
International Fax:
1-203-298-6402
E-mail via INTERNET:
object@cadkey.com

Q: What kind of training and support do you offer?

A: You can receive training through scheduled training seminars. Technical support will be through bulletin board access. Programmers responsible for developing CADKEY OBJECT Developer will answer your questions posted on CompuServe (Go CADKEY) and Internet (object@cadkey.com).

Q: Do my applications have to be compatible with Cadkey's applications?

A: No. Standalone application developers can create their product and utilize their own database structure. AFX developers can create applications which adhere to Cadkey's standard applications framework and database format.

Q: How does CODE support OLE/2?

A: This initial edition of CODE is not, as yet, configured as an OLE Automation Server- we plan to have this implemented for our next release. However, with Visual C++ Version 2.0 and MFC Version 3.0, supported by this edition of CODE, you can build your own OLE Client and OLE Container applications which use CODE objects, methods and functionality. If you wish to export CODE objects for OLE automation, you must create your own wrapper classes.

Cadkey, Inc., the Design and Engineering Productivity Company

Since 1981, CADKEY, has been committed to accelerating the productivity and creativity of mechanical design engineers around the world. CADKEY pioneered PC-based mechanical CAD by introducing CADKEY, the first truly three-dimensional, PC-based CAD software system in 1985. Available in 11 languages and honored with numerous awards, CADKEY systems are at work in Fortune 500 companies around the world.

Companies of every size have chosen CADKEY to handle their CAD needs, with more than 140,000 installations world wide. Over the last decade, CADKEY's distributor network has spread globally, with over 300 Value Added Resellers in North America alone.

Other Cadkey, Inc. Products:

CADKEY

CADKEY is an integrated 3D mechanical design, drafting and analysis software system that supports the complete design-to-manufacturing process. Cadkey contains many high-end 3D modeling features including construction planes, real time hidden line removal, shading and access to CADKEY ADVANCED MODELER and CADKEY ADVANCED IGES.

CADKEY DRAFTER

CADKEY DRAFTER is a powerful design and drafting product built for professional use. Drafter combines feature packed drafting and design with the ability to share data freely with other CAD systems.

DATA CAD

A sophisticated though easy to use 2D and 3D CAD program, that is easy to use, developed specifically for architectural, building, engineering and construction professionals.

Information in this document was accurate as of publication date. Cadkey, Inc., reserves the right to modify or enhance its products. CADKEY is a registered trademark of Cadkey, Inc. CODE, OBJECT Developer, and DataCAD are trademarks of Cadkey, Inc. ObjectStore is a registered trademark of Object Design, Inc. AutoCAD and DXF are registered trademarks of Autodesk, Inc. Microsoft is a registered trademark and Windows 3.1, Windows NT and the Windows logo are trademarks of the Microsoft Corporation. All other products are registered trademarks of their respective companies. © Copyright 1994, Cadkey, Inc. All rights reserved.