MICRO FOCUS®

extend®8
Interoperability Suite

Imagine. COBOL interoperating with Java™ technology... running a .NET™ assembly… processing XML data…or providing a Web service. Imagine running COBOL from a mobile, handheld device. What would this kind of functionality mean to your business? For some, it might give a competitive edge, adding value to mission-critical applications. To others it might mean breathing new life into a languishing application. ACUCOBOL-GT offers all of these capabilities and more.

ACUCOBOL-GT®
Open Systems COBOL Compiler and Runtime System

Overview

ACUCOBOL-GT is an advanced COBOL development system for extending and modernizing business-critical COBOL applications. It is an extremely portable COBOL, supporting single-source, compile-once deployment on hundreds of platforms including most UNIX®, Linux®, Windows®, VMS, and MPE/iX systems. ACUCOBOL-GT includes extensive facilities for interoperating with C, Java, .NET and XML. It supports connectivity to leading relational databases and ISAM data sources through both open and patented technologies. The core component of Acucorp's family of extend® technologies, ACUCOBOL-GT supports thin client and multi-tier client/server architectures, as well as COBOL Web services and platform-independent graphical user interfaces. Flexible and scalable, ACUCOBOL-GT works anywhere in the enterprise, from back office to point-of-sale.

ACUCOBOL-GT is a multi-component COBOL development system that includes a compiler, the COBOL Virtual Machine™ (runtime), the Vision indexed file system, a source-level interactive debugger and nearly a dozen support utilities. Our graphical technology allows you to create a native graphical user interface with the Screen Section and familiar COBOL verbs like DISPLAY and ACCEPT. A large set of portable runtime routines supports common operations such as executing operating system functions, retrieving error codes, managing files and managing memory. ACUCOBOL GT offers extreme portability. With the COBOL Virtual Machine, your compiled applications can be moved from one platform to another without changes or recompilation.

Development System

Essential development system features include:

Core technology

- Fast compiles to compact, machine-independent object code or, optionally, native object code for Intel®, PA-RISC®, PowerPC® and SPARC®. The code, compile, test cycle is more efficient because the compiles are fast and there is no link step.
- Machine-independent object code is portable to hundreds of platforms or operating systems.
- Program execution is managed by a high-performance, high-throughput COBOL Virtual Machine.
- Support for source compatibility with most popular COBOL dialects
- ANSI 1985 compliant with elements of the 1989 supplement and ISO/IEC 2002 standard
- Support for most ANSI 89 Intrinsic functions
- The native Vision indexed file system is scalable and performance-optimized (the RMS file system is used on VMS and OpenVMS platforms).
- An open file system interface provides broad data access to leading RDBMS, ODBC, ISAM and EXTFH-compliant data sources.

ACUCOBOL-GT®
Open Systems COBOL Compiler and Runtime System

COBOL extensions

- Native graphical user interface (GUI) technology for COBOL, including support for Windows native 32-bit controls, ActiveX controls, COM objects and .NET assemblies
- Extensive support for the Screen Section
- Support for programming and managing multiple execution threads (multi-threading)
- Support for transaction management
- A large set of portable library routines, including a routine that supports inter-process communication via sockets
- Support for external sort modules

Interoperability with other technologies

- Robust pathways to and from C and Java
- The ability to read, process and write XML data, either transparently or with direct control
- Support for COBOL-based Web services
- Support for Microsoft® .NET technologies
- Interoperability with J2EE application server platforms such as BEA WebLogic Server® and IBM WebSphere®
- The ability to interoperate with wireless devices such as mobile phones and Personal Digital Assistants

Deployment options

- Specialized, proprietary Thin Client technology
- Ability to deploy graphical applications from the Java command line or Eclipse
- Extensions for quickly and easily deploying Internet-based applications
- A high performance back-end client/server option
- The ability to create and deploy applications in a distribute processing architecture

A comprehensive development system

- A powerful, easy-to-use interactive, source-level debugger
- A sophisticated, easy-to-use integrated development environment for Windows: AcuBench® (licensed separately)
- An application profiling facility to help tune system performance
- The Abend Diagnostic Report facility to help analyze system failures
- Several specialized file handling utilities

Expert support and services

Acucorp is proud to offer worldwide customer service and technical support. Our professionals are experts in COBOL, Acucorp technology and best practices for deploying successful enterprise systems. They offer expert analysis, development and project management to the degree your business requires.
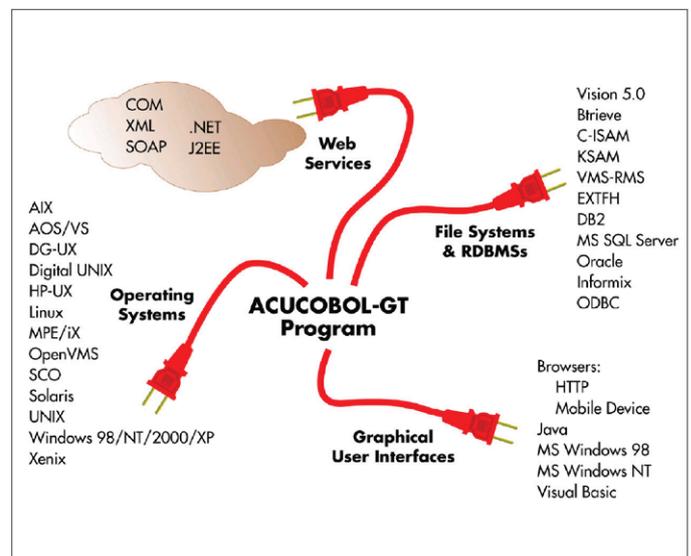

Figure 1: The ACUCOBOL-GT Environment Applications written in ACUCOBOL-GT "plug into" most modern enterprise platforms, file systems/databases, graphical user interfaces, Web service infrastructures and more.

The *extend*® Family of Solutions

The ACUCOBOL-GT development system is the core component of Acucorp's extend family of solutions. Other extend solutions include:

- AcuBench: Acucorp's Windows integrated development environment for ACUCOBOL GT
- AcuConnect®: client/server solutions for the strategic distribution of application processing and support for Acucorp's Thin Client technology. Thin Client technology allows you to display the user interface portion of your server-based application on a Windows client. Distributed computing support and Thin Client support are licensed separately.
- Acu4GL®: patented technology that supports transparent access from COBOL to leading relational database management systems (RDBMSs) and ODBC-compliant data sources

ACUCOBOL-GT®
Open Systems COBOL Compiler and Runtime System

- AcuXDBCTM, AcuXDBC Server, and AcuXDBC Enterprise Edition: a data management system designed to transform Vision files into a relational database-like environment, and provide access to Vision data from Microsoft Word, Excel, Access, MS Query, Crystal Reports® and other ODBC-compliant applications. JDBC connectivity is also available as well as remote access and SQL processing via AcuXDBC Server and AcuXDBC Enterprise Edition.
- AcuSQL®: embedded SQL support for ACUCOBOL-GT applications
- AcuServer®: client/server remote file access to Vision indexed, relative and sequential files as well as object files stored on UNIX, Linux and Windows servers
- AcuXUI™: a cross-platform user interface engine that allows graphical ACUCOBOL-GT programs to exhibit their user interface on UNIX and Linux platforms as well as Windows platforms.

extend® also includes the ACUCOBOL GT Web runtime and Web Thin Client, ActiveX versions of technologies designed to facilitate rapid Internet deployment of legacy applications.

Highlights

Core Technology

The ACUCOBOL-GT compiler is a fast and accurate single-pass compiler that is ANSI 1985 compliant with elements of the 1989 supplement and ISO/IEC 2002 standard. It generates object code from sources written in ACUCOBOL-GT and many common COBOL variants.

ACUCOBOL-GT generates compact, machine-independent object code or, optionally, native object code for Intel, PA-RISC, PowerPC, and SPARC processors. Once compiled, the machine-independent object code is portable to hundreds of platforms and several operating systems (see the section titled "Object Portability"). Object code is executed at run time by the COBOL Virtual Machine. Native-code objects are completely portable within their own family of processors. It is not necessary to relink after porting an object, whether the object is machine independent or native code. Native-code and machine-independent objects can be mixed within the same application, so you can compile your code to best suit the needs of individual programs, thus optimizing the performance of your complete application.

Source Code Compatibility

The ACUCOBOL-GT compiler provides several modes for increasing COBOL compatibility. These compatibility options are specified on the command line when each program is compiled. In addition to options that provide fine control over such elements as source file format, reserved word handling and data storage, ACUCOBOL-GT offers general compatibility with the following COBOLs:

- RM/COBOL-85 (ANSI 85)
- RM/COBOL version 2 (ANSI 74)
- ICOBOL
- VAX COBOL
- IBM DOS/VS COBOL
- HP COBOL II/XL

Different programs may use different compatibility modes, even if they are part of the same run unit. For example, if you have two programs, one written in RM/COBOL and the other in ACUCOBOL-GT, and you want to use them in the same application, the ACUCOBOL-GT compiler and runtime can handle this scenario.

Although ACUCOBOL-GT's compatibility options greatly simplify the conversion of existing applications to ACUCOBOL-GT, some changes to your code may be required.

Object Portability

Applications that you want to deploy on multiple platforms can be developed and maintained with a single set of source code. The source code needs to be compiled only once. The same object file can be delivered for use on any platform supported by ACUCOBOL-GT. For example, an application developed for UNIX can be run on a Windows workstation without recompilation. This is accomplished first by the compiler, which generates machine-independent object code, and second, by the COBOL Virtual Machine, which makes use of a host-specific runtime configuration file to provide definitions for machine-dependent values.

Machine-independent object code can be run on any of hundreds of platforms and several operating systems, including most common variants of UNIX and Linux (including Linux on the AS/400), 32- and 64-bit Windows systems (excluding Windows CE), HP MPE/iX (including the POSIX shell running on MPE), OpenVMS and others.

ACUCOBOL-GT is also available on the IBM System x5, p5, i5, and z9 family of servers running Linux, Windows, and AIX5L operating systems. This object portability makes ACUCOBOL-GT applications very appealing for Web deployments across a large range of platforms.

ACUCOBOL-GT®
Open Systems COBOL Compiler and Runtime System

Fast Native Indexed Files

The ACUCOBOL-GT development system includes the Vision indexed file system (except with VMS/OpenVMS where the native RMS file system is used). Vision is tightly integrated with all ACUCOBOL-GT components. It offers functionality comparable to other commercial indexed file systems, but with additional features and enhanced performance. Vision files may be created with either single- or dual-file format. The single file format includes both the data records and the overhead key information. The dual-file format separates data records from the record key information on the disk. This increases reliability when files must be rebuilt. It also facilitates additional performance-enhancing features, such as "geographic" proximity of data segments on the disk and "bulk" additions of records. A single Vision version 4 or 5 file has a logical limit of 128 terabytes of data.

Data Source Flexibility

ACUCOBOL-GT has an open architecture for connecting to other file systems. This means that in addition to Vision, you can choose from a variety of file systems and database management systems. Supported third-party ISAM file systems include Btrieve from Pervasive, KSAM and C-ISAM. If you add our Acu4GL technology, you can also access, via standard COBOL I/O statements—no SQL required—popular RDBMS data sources, such as Oracle®, Microsoft SQL Server, DB2®, Informix and Sybase as well as a variety of ODBC-compliant data sources. Using AcuSQL technology or a proprietary pre-processor, you can embed SQL in your COBOL programs, giving COBOL applications embedded SQL access to IBM DB2, Oracle, Microsoft SQL Server, MySQL and ISO/ANSI SQL92-compliant data sources. You can also access data through an EXTFH-compliant library.

In addition, ACUCOBOL-GT applications can read, process and write XML documents. For more information, see the section titled "Interoperating with XML".

COBOL Extensions

Graphical Technology

The GT in ACUCOBOL-GT stands for our advanced Graphical Technology. ACUCOBOL-GT includes a comprehensive set of COBOL extensions for programming and managing graphical user interfaces (GUI). With these extensions, you can add a full-featured, native GUI to an existing program entirely in COBOL and deploy the GUI on Windows or Java desktops. ACUCOBOL-GT's GUI capabilities include:
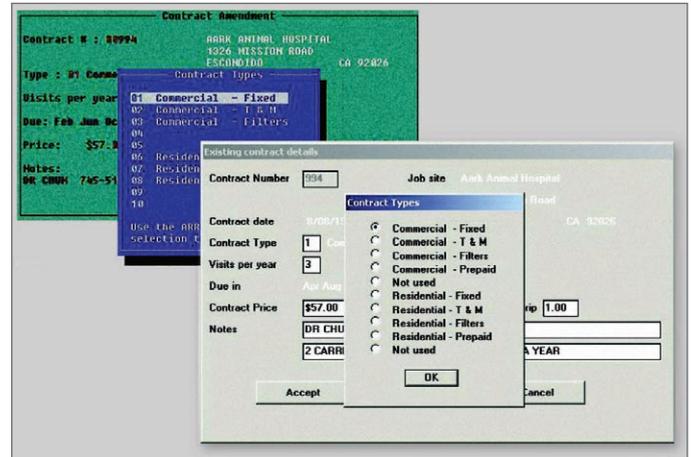

Figure 2: Graphical COBOL ACUCOBOL-GT lets you convert COBOL green screens into modern graphical user interfaces (GUIs) that your users expect.

- Native floating (movable) windows
- Menu bars and submenus
- Toolbars
- A wide variety of programmable GUI controls including: bar*, bitmap*, check box, combo box, date entry*, entry field, frame, grid*, label, list box, push button, radio button, scroll bar*, status bar, tab*, tree view and web browser* (Items marked with an "*" are not supported in text-mode environments.)
- Support for ActiveX controls, COM objects and .NET assemblies under Windows
- Access to the most common Microsoft controls under Windows
- Display of bitmaps and bitmap buttons under Windows
- Access to the native message box facility
- Access to the native file open and file save-as dialog boxes
- Access to the native help facility and support for context-sensitive help
- Specialized mouse handling
- Custom font selection and handling
- The ability to play ".WAV" audio files on Windows systems
- Custom colors
- Full-featured printing capabilities for Windows

ACUCOBOL-GT provides character-based emulation of most graphical controls as well as programming and runtime supports for graphically enhanced applications running in both graphical and character-based environments. ACUCOBOL GT supports applications that have a combination of character-based and graphical screen definitions and provides a mechanism for specifying which definition to use. This allows you to build a single user interface for use in both graphical and character environments.

ACUCOBOL-GT®
Open Systems COBOL Compiler and Runtime System

Fast Native Indexed Files

ACUCOBOL-GT includes extensive Screen Section capabilities that allow you to define an entire screen, including layout and field characteristics, in the Data Division of your COBOL program. Screen Section entries can refer to sections and paragraphs in the Procedure Division (called embedded procedures). Embedded procedures can be used for a variety of tasks, such as preloading data fields and performing immediate validation of user-supplied data.

Multiple Execution Threads

ACUCOBOL-GT allows you to specify and control multiple execution paths in a program. Each thread runs in and is managed by the current invocation of the runtime. The runtime monitors each thread and controls the amount of execution time each thread receives, switching among threads as necessary. Threads can have priorities assigned to them, external data can be shared among them, and threads may pass messages to one another. Multi-threading support is included in all runtimes and is internal to the runtime. It does not rely on or use any threading capabilities in the host operating system or the host system architecture. It is, therefore, completely portable across all supported platforms.

Transaction Management

ACUCOBOL-GT includes extensions to standard COBOL that allow you to implement a complete transaction management system. A transaction is a group of related file operations that are treated as an indivisible unit. The purpose of defining such transactions is to ensure that related files can be restored to a consistent state when errors occur. ACUCOBOL-GT includes special verbs (START, COMMIT, ROLLBACK) and a transaction logging facility for file operations that are part of a transaction. Once logged, the operations can be either committed or rolled back (undone) by the program.

Portable Runtime Library Routines

ACUCOBOL-GT comes with more than 40 fully portable runtime library routines. These routines provide total cross-platform support for such functions as dynamic memory management, common operating system file operations (COPY, RENAME, MOVE, etc.), mouse handling, error code recovery and more. One particularly valuable routine provides support for inter-process communication via sockets.

External Sort Modules

ACUCOBOL-GT applications that are deployed on UNIX and Linux systems can be configured to use an external, third-party sort module, such as SyncSort® or CoSORT™, to handle SORT and MERGE operations. In some cases, using a specialized sort module allows you to specify unusual operations or to achieve superior performance. The external sort capability is enabled through the EXTSM interface.

Facilities for Interoperating with Other Languages and Technologies

Interoperating with C and Java

In many enterprise systems, C and Java programs provide core capabilities and support functions. To help you build more integrated applications that make the best use of COBOL programs, ACUCOBOL-GT provides several robust methods for interoperating with C and Java programs.

To support the calling of C from COBOL, ACUCOBOL-GT allows:

- Direct calls to C programs located in Windows DLLs and UNIX/Linux shared object libraries
- Direct and "interface" calls to C programs linked into the ACUCOBOL-GT runtime

To support the ability of C programs to call COBOL, ACUCOBOL-GT includes an extensive C Application Program Interface (API).

To enable Java programmers to call COBOL, ACUCOBOL-GT provides a Java class known as CVM that encapsulates the COBOL Virtual Machine (runtime). With the CVM, the Java programmer can programmatically instantiate an instance of the ACUCOBOL-GT runtime to invoke a COBOL program. The programmer can use other classes or methods of the CVM to specify runtime and program options. Alternatively, you can use Java compiler options to automatically generate Java classes that can call your COBOL program.

To support calls to Java from COBOL, ACUCOBOL-GT provides a library routine called C$JAVA. Calling C$JAVA causes the Java Virtual Machine (JVM) and the specified Java class to be loaded. You can then create a Java object, call the methods of a Java object, create and use Java arrays, use Java logging and more.

Interoperating with XML

ACUCOBOL-GT allows applications to read, process and write XML documents, wherever they reside. The built-in AcuXML interface allows ACUCOBOL-GT applications to process XML documents like regular sequential files.

ACUCOBOL-GT®
Open Systems COBOL Compiler and Runtime System

The AcuXML interface also converts COBOL output data into XML format as required. For situations that require precise control over the XML data, ACUCOBOL-GT includes a runtime library routine called C$XML. Using C$XML you can retrieve, parse, modify and write XML files.

Interoperating with ActiveX, COM and .NET

ACUCOBOL-GT supports the use of ActiveX controls, COM objects and .NET assemblies. When you add one of these objects to your program, it becomes part of the development and runtime environment, providing the application with new functionality. By supporting ActiveX, .NET and COM, ACUCOBOL-GT allows you to take advantage of existing software functionality as well as create applications that conform to the latest Windows standard.

There are several methods for calling a COBOL program from .NET:

- .NET compiler options automatically generate either a .NET executable file or a dynamic link library. .NET programmers can invoke these objects as they would any native .NET object.
- Native .NET interface enables programmers to start and control a COBOL program at the API level.
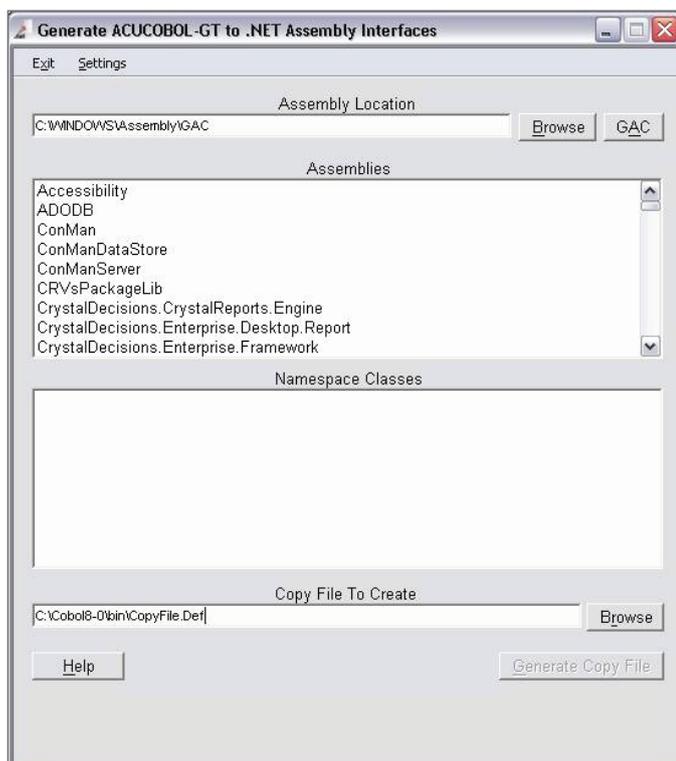
Figure 3: .NET and ActiveX Definitions Generator ACUCOBOL GT includes utilities that generate COBOL FDs and SELECT statements from .NET assemblies and ActiveX controls.

Providing and Consuming Web Services

As part of our continuing commitment to legacy Information Technology extension and interoperability, Acucorp offers a variety of ways to work with Web services. For those using the J2EE platform, Acucorp offers a native Java interface. By invoking the Java class contained in this archive, a Web service running on J2EE can start the runtime and run your COBOL program.

For those using .NET, Acucorp offers a .NET interface that presents the runtime as a dynamic link library. By invoking this DLL, a Web service running on .NET can start the runtime and run your COBOL program.

To enable a COBOL program to consume a Web service, Acucorp provides the C$JAVA library routine and the NETDEFGEN utility.

Interoperating with Wireless Devices and Personal Digital Assistants

Acucorp offers several paths to interoperating with Wireless Access Protocol (WAP) devices such as Personal Digital Assistants (PDAs) and mobile phones.

- ACUCOBOL-GT COM Server
- ACUCOBOL-GT Common Gateway Interface (CGI) language extensions
- ACUCOBOL-GT runtime and Short Message Service (SMS) processing

Deployment Options

Thin Client

IACUCOBOL-GT programs can be easily integrated with Acucorp's Thin Client technology. This technology allows ACUCOBOL-GT programs running on UNIX, Linux or Windows servers to present a full Windows GUI on Windows PCs networked with TCP/IP. These applications enjoy the benefits of centralized maintenance and the performance characteristics of a thin client architecture. Many applications perform better when deployed in a thin fashion compared to other networking techniques, because COBOL programs execute on the server where the data is local.

Thin client technology consists of three key components. First, a small program on the Windows client communicates with the application running on the server and handles the display of the user interface. This thin piece is called the ACUCOBOL GT Thin Client. It can be downloaded at no cost from the Acucorp Web site. Second, there is a listener service running on the UNIX,

ACUCOBOL-GT®
Open Systems COBOL Compiler and Runtime System

Linux or Windows server called AcuConnect. AcuConnect listens for requests from thin clients and launches the third component, a standard ACUCOBOL-GT runtime. Once the application has started, the ACUCOBOL-GT runtime communicates directly with the thin client, and AcuConnect returns to its role of listening for thin client requests. See the AcuConnect technology datasheet for more information. AcuConnect is a separately licensed product.

extend® includes both ActiveX and DLL versions of the Thin Client technology, enabling users to call thin client applications from Web browsers and other client environments.

Internet Deployment

extend® and ACUCOBOL-GT include a variety of technologies that simplify the deployment of ACUCOBOL GT applications on the Internet. By applying these technologies, you can quickly reconfigure an application that is currently running in a silo to provide access to remote clients, distributors or employees without cumbersome changes to the program.

For example, you can use an ActiveX version of the ACUCOBOL-GT runtime to quickly deploy your COBOL applications on the Internet with few or no changes to the source code. The ActiveX version is known as the ACUCOBOL-GT Web Runtime.

If you prefer the footprint of a thin client architecture, you can use ACUCOBOL-GT Web Thin Client, an ActiveX version of the ACUCOBOL-GT Thin Client. The Web Thin Client makes existing thin client applications immediately available on the Web, and provides all of the benefits of the standard Thin Client, including GUI options for UNIX, centralized maintenance and exceptional performance.

Both the Web Thin Client and Web runtime are designed to be automatically downloaded and installed on end users' PCs when they visit your Web page. They can also be installed from ACUCOBOL-GT media or downloaded from the Acucorp Web site. They are distributed in CAB files with a digital signature from Acucorp, giving users confidence that they are safe to use on their computers. With the Web runtime, your application automatically runs on the user's local machine, but with the Web Thin Client, it runs on the server, projecting just the user interface on the client, as in a standard thin client configuration.

ACUCOBOL-GT also includes extensions that make it easy to write Common Gateway Interface (CGI) scripts. Programs can be made to read CGI input data, process it and output response data in HTML, WML, XML or other MIME Content Types. This enables ACUCOBOL-GT programs to interact with Wireless Application Protocol (WAP) devices like mobile phones and PDAs as well as standard HTTP Web browsers. In addition to providing

cross-platform compatibility, ACUCOBOL-GT CGI programs do not require any special plug-in or applet when interfacing with a browser.

Java-based GUI Deployments

With AcuXUI, an add-on component, you can run your graphical COBOL program from the Java command line or Eclipse. Graphical controls such as windows, entry fields, and radio buttons are described in the COBOL program with ACUCOBOL-GT. Rather than directing the Windows operating system to create the controls, the runtime directs the Java Runtime Environment (JRE) to create the controls on a Java desktop. For this reason, the controls can run on most operating systems, including UNIX, Linux and Macintosh. The Java desktop can also run on Windows.
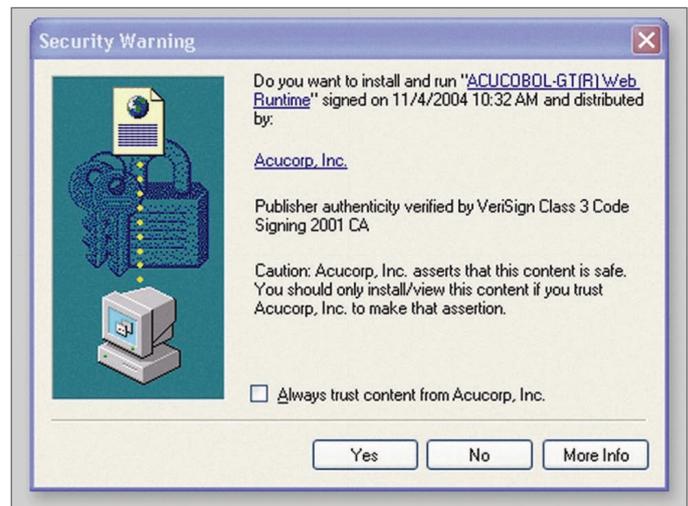

Figure 4: Global Access to a COBOL Runtime The Web runtime and Web Thin Client are ActiveX versions of the ACUCOBOL-GT runtime/Thin Client that are automatically downloaded from the Web to start your COBOL program.

Back-end Client/Server Deployments

ACUCOBOL-GT applications can easily acquire full back-end client/server support for all Vision indexed, sequential and relative files with AcuServer. Through AcuServer, you can store data files on any UNIX, Linux or Windows server and access them from any UNIX, Linux or Windows client. Network file access is performed with full record locking support. See the AcuServer technology datasheet for more information.

Distributed Processing Architectures

Our AcuConnect technology enables ACUCOBOL GT applications running on client machines to launch applications on server machines connected via TCP/IP (see the AcuConnect technology

ACUCOBOL-GT®
Open Systems COBOL Compiler and Runtime System

datasheet for more information). The server can reside on a local area network, wide area network or the global Internet. AcuConnect is a separately licensed product.

A Comprehensive Development System

Interactive Debugger

Built into the ACUCOBOL-GT runtime is a full-featured, easy-to-use interactive debugger. When activated, the debugger runs in a separate window that overlays the running program without interfering with the application's normal screen I/O. The debugger operates in three distinct modes: source, symbolic and low-level. All modes include support for the Trace Files command, which can be used to trace file I/O and other application functions. Source debugging requires that the program be compiled with a special switch and allows the programmer to view and interact with the COBOL source code while the application is executing. Application source code is displayed in the debugger window, and the programmer can interact directly with the source code to set breakpoints, inspect variables, execute the program one line of code at a time and more.

Integrated Development Environment

On Windows systems, ACUCOBOL-GT can be paired with the AcuBench integrated development environment (see the AcuBench technology datasheet for detailed information). AcuBench provides access to the ACUCOBOL-GT compiler and runtime system with a set of graphically-based development tools, including project management facilities, a drag-and-drop WYSIWYG screen designer, a language-sensitive source code editor and a graphical debugger capable of remotely debugging applications executing on Windows, UNIX or Linux environments. AcuBench is a separately licensed product.

Remote Preprocessing

The **Boomerang** utility program includes client/server technologies that enable you to send source and COPY files to a remote server, invoke and perform preprocessing on that server, then return the processed file to your client machine where additional compiling can occur. Many proprietary preprocessors, such as those from Oracle and IBM, have machine-specific functions that require preprocessing to occur in their native environments. **Boomerang** makes accessing these preprocessors easier and more efficient.

Application Profiler

To help with the critical task of tuning application performance, ACUCOBOL-GT includes a runtime profiling facility. When you enable profiling, the runtime monitors the application during execution and collects statistics on where the application is spending its time, including I/O operations and CALLs. You can then run a utility called acuprof on the resulting data file to parse the information and produce a performance report. This report can help you pinpoint areas of possible improvement.

Abend Diagnostic Report

The runtime can be configured to produce a report that describes the state of a program at the moment that an abnormal shutdown occurs. This is called the Abend Diagnostic Report, or ADR. This report is helpful in performing post-mortem analysis of the cause of a shutdown. The report contains three major sections:

- General information about the program, such as the command-line parameters, the reason for the shutdown and the line number where the shutdown occurred
- A call stack summary of each thread
- Detailed information about each program, including all the data item values

File Handling Utilities

cblutil is a utility for working with ACUCOBOL GT object files. **cblutil** has the ability to place object files and resources such as bitmaps together to create object libraries. It also allows you to print information about an object file or object library. And it can be used to generate native-code object modules from machine-independent object files.

vutil is a file utility for working with Vision indexed files. **vutil** allows you to examine files, extract data records, change the maximum record size and rebuild corrupted indexes. The functions are designed to allow you to specify all possible task parameters up front, so that the utility can run unattended or with a minimum of user interactions.

vio is ACUCOBOL-GT's file transfer utility. **vio** allows you to collect files together into archives, and to extract files from archives. Typically, an archive is placed on some external media such as a tape or a diskette, but the archive may also be a disk file. **vio** is often used to back up a set of files or to move files from one machine to another.

ACUCOBOL-GT®

Open Systems COBOL Compiler and Runtime System

AcuSort is a utility program that lets you perform external sort and merge operations on Vision indexed, relative, binary, sequential and line sequential files. In some cases, AcuSort offers better performance than the SORT verb. This utility is invoked from the command line where instructions can be passed directly or through a text file.

Development System Support and Services

ACUCOBOL-GT is a sophisticated and versatile development system. To help you realize its full potential and achieve the results you want and need, we have a worldwide, world-class team of knowledgeable, responsive customer service, technical support, and professional services people. Our professionals know COBOL, are experts in Acucorp technologies, and practice the art and science of deploying successful enterprise systems.

Platform Support

Windows
- MS Windows XP, 2000, NT, Vista
- MS Windows 2003 Server, 2000-Advanced Server, 2000 Server, NT Server
- MS Windows Terminal Server/Citrix

UNIX
- HP-UX 11.0, 11i, and 11.22i (32-bit and 64-bit)
- SCO UNIX Open Server (5.0.6) ELF
- SCO UNIX V.3 R2.4
- SCO UnixWare 7.0 ELF
- UNIX SVR4v3 (MP-RAS 3.x)

Linux
- Linux (glibc 2.0, 2.1, 2.2, 2.3, 2.4, 2.5) - x86 PC (Intel)
- Linux (glibc 2.1, 2.2, and 2.3) - IBM (S/390)
- Linux (glibc 2.2) - IBM System i5
- Linux (glibc 2.3) - IBM System p5

AIX
- AIX 5L (32-bit and 64-bit)

Solaris
- Solaris 8.0, 9.0 (Sun SPARC, 32-bit and 64-bit)
- Solaris 8.0 (x86 PC/Intel, 32-bit)
- Solaris 10

VMS
- OpenVMS operating system

MPE/iX
- MPE/iX 6 (HP e3000)

Native Code—Supported Processors
- Intel: Intel 386, 486, the Pentium family (and compatible)
- PA-RISC: 32- and 64-bit processors under HP-UX and MPE/iX
- PowerPC: 32- and 64-bit IBM System p5 under AIX
- SPARC: SPARC v7, v8, v9 (and compatible)

ANSI Module Implementation Levels
Nucleus (1-2)	Level 2
Sequential I-O	Level 2
Relative I-O	Level 2
Indexed I-O	Level 2
Inter-Program Communication (1-2)	Level 2
Sort-Merge (0-1)	Level 1
Source Text Manipulation (0-2)	Level 2
Segmentation (0-2)	Level 1
Report Writer (optional)	not supported
Communication (optional)	not supported
Debug (optional)	not supported