



API Connectivity Data Sheet

Ardent UniVerse RDBMS offers a complete family of products to meet today's key requirements in building client/server and web-enabled applications — fast application development and ease of integration throughout the enterprise. These include toolkits that enhance productivity and speed application development and allow seamless sharing of data among applications. These products work with UNIX and Windows NT-based UniVerse servers.

Technologies

The UniVerse portfolio of technologies provides developers with a comprehensive set of tools that make a wide variety of tasks faster and easier, while also allowing them to add a full range of impressive capabilities to their open systems environments.

These products fully support the ability to use a full range of desktop development tools to build new UniVerse applications or enhance and extend existing ones. They make it easy to integrate data from virtually any popular database, including Oracle, Sybase and Informix. And they help provide seamless access to and from UniVerse and support for Internet or Intranet applications.

UniVerse technologies fall into two categories, depending on the major goal they support:

- Application Development (Windows or browser-based)
 - UniVerse Objects
 - UniVerse ODBC
 - UniVerse Call Interface
- Enterprise Integration
 - UniVerse ODBC
 - General Call Interface

The following discussion is designed to provide an overview of the features of each technology, plus help developers select the best tool for each task.

Developing Applications for Windows and the Web

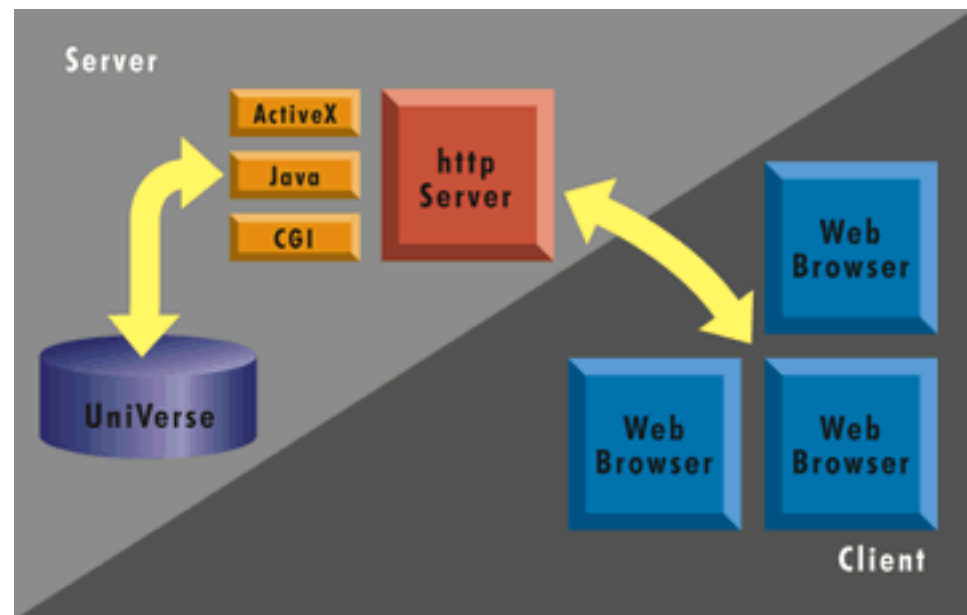
UniVerse offers three levels of Application Programming Interfaces (API): UniVerse Objects, UniVerse ODBC, and UniVerse Call Interface.

◁ UniVerse Objects ▷

Because of its comprehensive capabilities and rich functionality, UniVerse Objects (UV/Objects) is the preferred API for UniVerse and is the best choice for most application development programs and projects.

UV/Objects is an ActiveX control (OCX) whose objects are able to take full advantage of the dual interface standard for either fast COM or flexible automation interfaces, both supported by Visual Basic. As a result, UV/Objects brings the proven advantages of object-oriented development to UniVerse client/server or Web development. The introduction of UniVerse National Language Support (UV/NLS) capabilities also provides developers with a unified method for creating applications for use in any country in the world. In the case of Visual Basic, or Visual InterDev using VB Script, UV/Objects becomes a natural extension to the language.

The objects provided are: Session, File, Dynamic Array, Select List, Dictionary, Command, Sequential File, Transaction, NLSMap, NLSLocale, and Subroutine object which permits reuse of existing application subroutines. All objects can be manipulated in the same rapid and easy-to-use manner as a traditional UniVerse application.



UV/Objects supports rapid application development. Records are read and written using “Dynamic Array” objects, which provide easy-to-use access to complex data structures representing real-world business objects such as orders, invoices, customers, etc. Its ability to fully support the reuse of components also helps ensure the development of high quality applications faster, more efficiently, and often at a lower cost.

UV/Objects lets companies capitalize on using Microsoft COM technology for new GUI application interfaces. Yet they also leverage today’s UniVerse application by calling existing UniVerse catalog subroutines. Developers can easily change the front end, without sacrificing years of investment in existing technology.

⌂ UniVerse ODBC ⌂

While UniVerse ODBC (UV/ODBC) is primarily an enterprise integration and desktop connectivity tool, it can be used as an API to develop applications using SQL, rather than native UniVerse record access. It is a logical choice for users of PowerBuilder, Microsoft Active Data Objects, or OLE DB who wish to

use UniVerse. It is also an excellent tool for UNIX ODBC-oriented application development using C, Java (using a JDBC/ODBC bridge), or Netscape's LiveWire. In addition, UV/ODBC is an excellent tool for providing integration between another RDBMS and UniVerse.

UV/ODBC supports the use of UniVerse tables, views and files with D, S, A and I dictionary types. This eliminates the need to make UniVerse data files appear more SQL-like in order to facilitate high-performance ODBC read and write access. UV/ODBC also takes full advantage of UniVerse multivalued data structures and extended features, such as select lists.

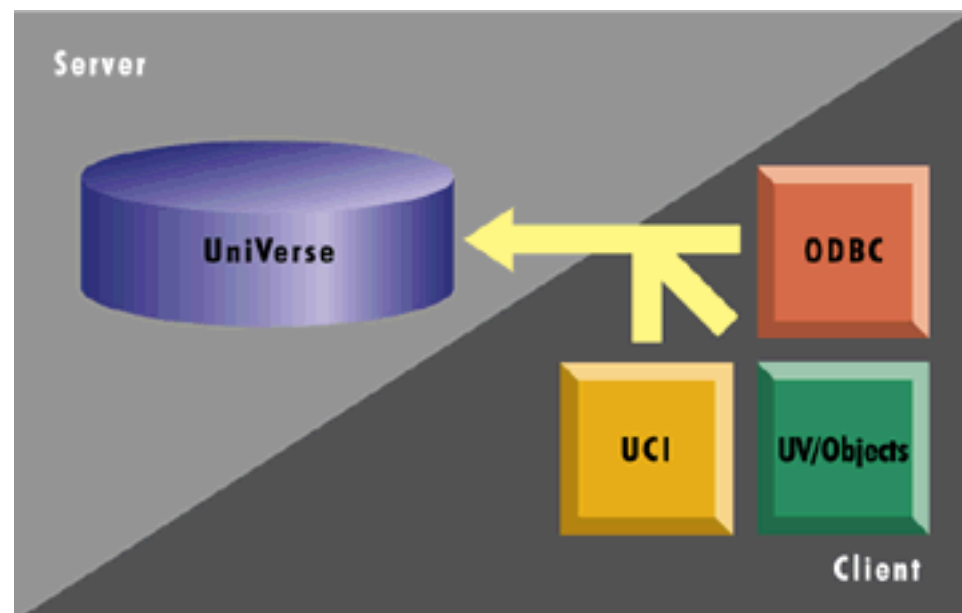
With UniVerse ODBC, UniVerse users can implement an ODBC solution today, and gradually migrate to using the full UniVerse SQL interface, with its security and declarative integrity features.

An ODBC interface provides true database independence since UniVerse features appear transparent and are treated like any other RDBMS. However, because it uses a relational, set-oriented approach, it is more alien to the traditional UniVerse developer than a record-oriented approach used by UV/Objects.

⌈ UniVerse Call Interface ⌋

The UniVerse Call Interface (UCI) is a 'C' language, UniVerse-specific SQL Call Level Interface (CLI) that is available on all UniVerse platforms. Its purpose is to provide a high-performance, ODBC 2-based interface to UniVerse for developers and third-party tools vendors.

The UCI is available as a UNIX static library and a 32-bit Windows DLL. It uses the UniVerse RPC mechanism, permitting applications direct access to the UniVerse database for both local and networked UNIX or Windows NT-based UniVerse servers. It also provides direct access to all schemas/accounts, tables, views and files, and dictionary information using the UniVerse SQL engine.



The UCI has two modes of operation. The 1NF mode offers capabilities for strict SQL access to tables and views via the information catalog. The NF2 mode provides access to schemas and accounts via the Vocabulary (VOC) and UV.ACCOUNT files. While it does use ODBC-like primitives, it does not replace the use of UV/ODBC for ODBC-based enterprise and desktop integration requirements.

Choosing the Right API

Your choice of an external interface to develop applications and access UniVerse depends heavily on what you are trying to do. See Table 1 for an overview of the features and benefits of each. However, for most development environments and tasks, UV/Objects will yield the best results and is the preferred programming interface.

Select UV/Objects if you:

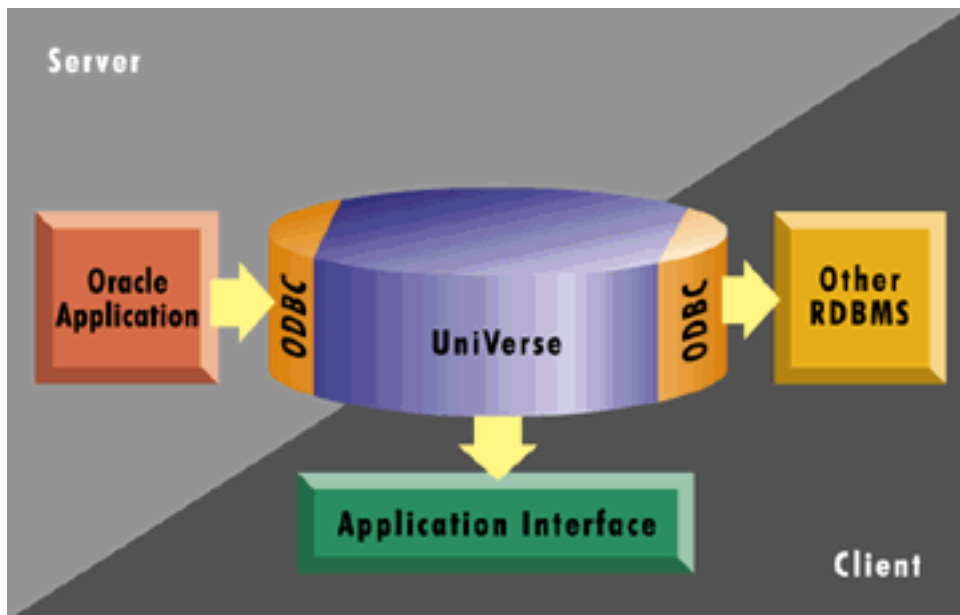
- want to develop modern Windows or browser-based GUI applications using standard Windows programming tools (e.g., Visual Basic, Visual InterDev and other ADE),
- have a substantial existing investment in product or programming skills in BASIC applications and want easy access to modern client/server or Web-enabled GUI front-end capability without a total rewrite, or
- want complete access to the proven benefits of an object-oriented development environment with its ability to reuse subassemblies (e.g., increased productivity and efficiency, lower costs, etc.).

Choose UV/ODBC if you:

- are primarily interested in using ODBC-aware, Windows productivity applications (e.g., Microsoft Office suite) or development tools (e.g., PowerBuilder, Microsoft ADO or OLE DB) with UniVerse.

However, you generally must be prepared to:

- learn new SQL skills
- lose some of your investment in existing BASIC programs
- are primarily interested in developing applications that are portable across multiple databases, with lowest common denominator functionality, or
- want to develop modern Windows GUI front-end applications (e.g., in Visual Basic, Visual InterDev, Visual C/C++ or other ODBC-enabled tools) and programs primarily at the SQL level.



Select UCI if you:

- are developing 32-bit Windows or UNIX-based applications and want to access UniVerse data at a raw, SQL level without the support of a full ODBC driver, or
- want access to the native UniVerse data model, either locally or on multiple machines, for developing applications with maximum flexibility.

If you want to distribute your application function so that user interface processing runs on the front-end client machine or within the browser, and data-intensive processing runs the back-end database server, then you can use any of these interfaces. For Windows, you can choose UV/ODBC if you will primarily use SQL-level access. For UNIX application development, use UV/ODBC or UCI. For Web-based development, use UV/Objects or UV/ODBC.

With UniVerse, its easy to select the right API for the right development situation.

API	Feature	Benefit
UV/Objects	<ul style="list-style-type: none"> - Supports object-based client/server and Web-based application development - Supports rapid Windows application development using popular desktop development tools 	<ul style="list-style-type: none"> - Exploit capabilities of standard Windows tools - Develop new applications using existing code and skills sets - Enjoy the proven advantages of object-oriented development
UV/ODBC	<ul style="list-style-type: none"> - Supports SQL access method 	<ul style="list-style-type: none"> - Develop applications using PowerBuilder - Provides advantages of a full ODBC driver for SQL development - Allows standard Windows productivity tools to access UniVerse's complex data structures - No need to "SQLize" the database for high-performance ODBC access

UCI	<ul style="list-style-type: none"> - Provides standards-based low-level interface to UniVerse for client/server development tools 	<ul style="list-style-type: none"> - Used for systems-level work - Fully exploits UniVerse in either unnested 1NF mode or nested NF2 mode
-----	--	---

Integrating the Enterprise with Ease

UniVerse offers three choices for seamlessly integrating other applications within the enterprise: UniVerse SQL Client, UniVerse ODBC, and General Calling Interface.

◁ UniVerse SQL Client ▷

UniVerse SQL Client (UV/SQL Client) provides an interface to other ODBC-accessible data sources from UniVerse. An ODBC programmatic interface, UV/SQL Client enables you to develop applications that can communicate with other UniVerse applications or any database with an attached ODBC driver (including Oracle, Informix, Sybase, etc.) locally or remotely.

Using a programmatic interface and suitable middleware integrates applications and databases in real time. This removes the need for cumbersome bulk data exchange mechanisms where no database interoperability mechanism exists. So, for example, a UniVerse-based manufacturing application not only reads and extracts data from a financial application, but manipulates and updates it, as well. Applications developed with UV/SQL Client have a multitude of potential database and platform connections, greatly extending the reach of UniVerse.

◁ UniVerse ODBC ▷

UniVerse ODBC is an ODBC driver that provides access to UniVerse from any 16- or 32-bit Windows desktop or UNIX server. When providing client-to-server connectivity, UV/ODBC allows developers to mix and match a wide variety of front-end applications with a full range of back-end databases. As a result, products like Excel and Access can connect to UniVerse data without losing the performance and flexibility advantages of their inherent data models.

In server-to-server connectivity, UV/ODBC can be used with the enterprise gateways built into most popular RDBMS's (e.g., Oracle's Transparent Gateway). This permits the client RDBMS direct access to UniVerse data from another database's application.

◁ General Calling Interface ▷

The General Calling Interface (GCI) is a call level interface that allows UniVerse subroutines (stored procedures) to call external functions such as OS routines, mail services, or non-intelligent device drivers. It offers a "lowest common denominator" approach, facilitating application-to-application connections.

Through GCI, UniVerse developers can exploit external Windows DLLs, UNIX applications, APIs, or RPC mechanisms using this intrinsic UniVerse functionality. Additionally, for ease of configuration and maintenance, the GCI interface is centrally configured and maintained.

UniVerse offers choices for seamlessly integrating other applications within the enterprise.

API	Feature	Benefit

UV/SQL Client	Intrinsic ODBC-like BASIC programmatic interface provides access to UniVerse and ODBC-configured data sources	Allows UniVerse applications to have real-time access to Oracle, Sybase, UniVerse, etc.
UV/ODBC	ODBC driver that provides access to UniVerse	Allows UniVerse to be simultaneously accessed by a variety of database gateways, applications and client platform combinations, UNIX and Windows
GCI	Offers "lowest common denominator" approach to enterprise integration	Can communicate with any application interfaces that may look "alien" to other tools.