

# The Alchemy<sup>®</sup> ODBC Interface

## An Open Interface for ad-hoc Query of Documents in the Alchemy Database

**Abstract:**

This paper discusses key concepts regarding the ODBC interface:

- Industry standard application access
- Why ODBC is important
- ODBC is an interface and a language
- User scenarios



**Information Management Research**

6025 South Quebec Street, Suite 260  
Englewood, CO 80111  
303-689-0022

[www.imrgold.com](http://www.imrgold.com)

Summer 2003 update for Release 7

# 1. Introduction

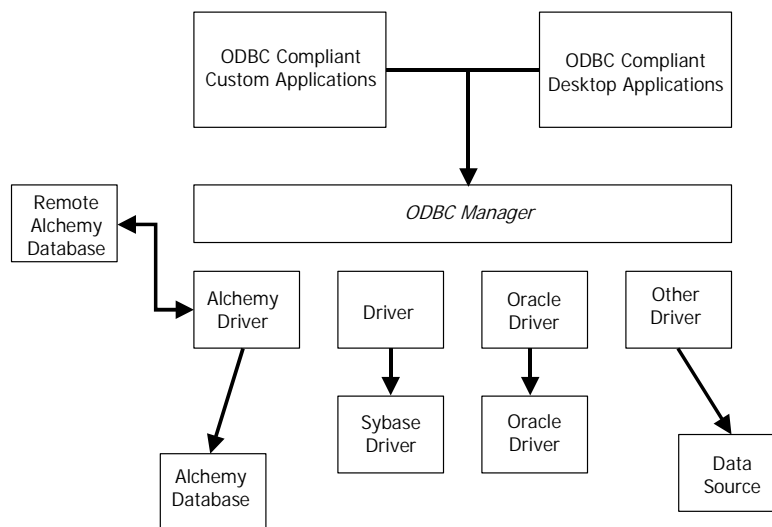
This document describes IMR's support for ODBC through its Alchemy v7.4 ODBC driver. An overview of ODBC and business objectives for this API is presented.

# 2. Overview

The following are key points within this document:

- Open Database Connectivity (ODBC) is an interface for accessing data in a heterogeneous environment of relational and non-relational database management systems.
- IMR's commitment to ODBC enables Alchemy databases to be quickly integrated with existing corporate application tools through an open, industry standard. Alchemy version 7.4 is delivered with an ODBC driver, supporting single client access to Alchemy databases. It enables report writing and ad-hoc querying of documents and Alchemy profile data using commercially available data access tools, development tools and applications.
- Alchemy ODBC support enables the use of development tools such as Sybase PowerBuilder, Microsoft Visual Basic, Visual C++ and .NET languages to create Alchemy-based archival applications. These development tools view Alchemy as just another database, similar to Microsoft Access, through their ODBC access interface.
- Alchemy ODBC support enables integration of other ODBC compliant applications such as Microsoft Word, Excel, Query, Access and Crystal Reports for report writing and ad hoc querying.

The following diagram presents Alchemy in the context of a typical ODBC environment:



### 3. Business Requirements

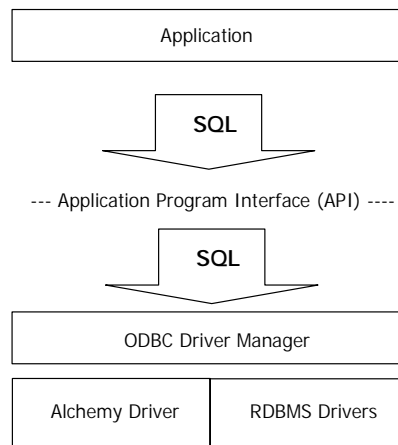
Corporate developers and IS managers with archival requirements wish to solve these problems:

- Very little time and budget are available to get new applications implemented or to learn new tools. High level tools are needed to meet development schedules.
- There is increasing demand to build applications that work transparently across a wide variety of databases. The corporate developer is looking for tools that hide the details of database differences from the application.
- Developers must be able to use in-house tools and expertise to build custom archival applications.

IMR's Alchemy 7.4 responds to these requirements by making its powerful archival engine look like an ODBC-enabled database environment to existing corporate applications and tools.

### 4. ODBC Overview

The following diagram shows a high-level picture of an ODBC environment:



#### 4.1 Why ODBC?

Since ODBC was introduced, it has gained broad support throughout the database industry. To date, every major database and tool vendor (including Oracle, Sybase, Informix, Computer Associates, IBM, Gupta, Borland, and Microsoft) and more than 60 leading application software and value-added resellers have added ODBC support to their products.

There are hundreds of ODBC-compliant applications and drivers available for the most popular data sources.

With 16- and 32-bit platform support under development for multiple UNIX versions, Apple Macintosh, Power Macintosh, OS/2, Microsoft Windows, ODBC continues to be an industry-wide standard.

ODBC addresses the database connectivity problem by using one common industry-standard application programming interface (API). ODBC is standard, open and vendor-neutral. ODBC allows application developers to use the same interface (API) and access technology across diverse platforms.

Applications connect to the database and retrieve data by making calls through the ODBC API to Microsoft's ODBC driver manager. The ODBC driver manager dynamically determines which server type is needed (e.g. Microsoft Access or Alchemy) to access the specified data source and invokes the appropriate ODBC driver for that database. In general, the application need not be aware of, nor understand the details of the differences between various data sources, this is handled by the ODBC driver manager. An application written to the ODBC API can substitute back end engines without requiring any code changes or relinking (assuming equivalent functionality in the new engine and driver). There is a provision for applications, which wish to take advantage of "extra" capabilities of a particular engine to get at this functionality through extensions in the ODBC driver. Within the context of an ODBC environment, there are two component types which must be ODBC-aware: applications which leverage the ODBC approach to obtain access to a variety of data sources and ODBC drivers for engines that actually access or manage these data sources. Microsoft has released applications, which utilize the ODBC architecture including Word, Excel, Query, and Access. Microsoft development tools allow access to ODBC data sources via direct access to ODBC or through other technologies such as ADO and ADO.NET. 4GL development tools and report writers supporting data access through ODBC drivers are available from other software vendors. Most relational database suppliers supply ODBC drivers for their engines. Various companies are also supporting access to flat files and spreadsheets through ODBC drivers as well.

## **4.2 ODBC Specifies an Interface and a Language**

The ODBC specification covers both the API and the dialect of SQL that will be used by the application to communicate with the server. For each of these, three levels of compliance are defined.

For the API, these are called Core, Level 1, and Level 2. For the SQL language, the levels are Minimal, Core, and Extended. An application can ask a given ODBC driver what level it supports and configure itself accordingly. Similarly, each driver must be able to respond with its compliance level.

In essence, the lowest levels of compliance are the least common denominator of support for all ODBC drivers. An application written to this level can be guaranteed to work with any ODBC driver (which could allow data to be stored in anything from flat files to complex database systems). At the other end of the spectrum, the Level 2 API provides a uniform method of accessing advanced features not yet provided by all relational databases. Relying on Level 2 features would be done at the expense of interoperability with other data sources. There are API functions, which are not relevant to read-only archival applications (such as transaction commit and rollback). Conversely, there is no support for some full text search and retrieval functionality such as search term highlighting.

## **4.3 ODBC Platform Support**

The ODBC driver manager is available for Microsoft for Windows 98 and Windows NT, Windows 2000, Windows XP and Windows 2003 Server.

# **5. Alchemy and ODBC**

IMR has identified support for ODBC-enabled applications and tools as a strategic objective as part of an overall theme of software that fits within corporate information architectures. Alchemy is delivered with an ODBC driver, supporting integration with a number of industry leading report writing and ad-hoc query tools for decision support applications.

## 5.1. *Alchemy ODBC Driver*

*The Alchemy version 7.4 is an ODBC 2.50 Core Level compliant driver. Support is limited to this subset because the underlying Alchemy database engine is not relational. However, the ODBC driver opens up a powerful interface to Alchemy.*

*Alchemy databases created using Alchemy v3.0 or later are supported.*

- (i) reads: regular SELECT statements with or without extended predicates, prepared SELECT statements without extended predicates (note1), SELECT statements with compound predicates (see limitations).
- (ii) insertions: regular INSERT statements subject to database triggers described above (In particular, see above, on how the Full\_Text and Full\_Binary columns are automatically updated on insertion.), prepared INSERT statements
- (iii) updates: regular UPDATE statements with or without extended predicates, prepared UPDATE statements without extended predicates
- (iv) deletes: regular DELETE statements with or without extended predicates, prepared DELETE statements without extended predicates
- (v) table joins: (This is an inefficient operation as Alchemy's heirarchical organization does not translate well to a table join).
- (vi) sorted reads: SELECT statements with GROUP BY, ORDER BY. . . clauses
- (vii) table modification: (ALTER TABLE) -- unsupported
- (viii) table creation: -- unsupported
- (ix) index creation: (CREATE INDEX) – unsupported

## 5.2 *User Scenarios and Pre-requisites*

Alchemy's support for ODBC can be leveraged in one of two ways, first as a means to generate reports based on information archived in Alchemy databases, and second, as a means through the ODBC API to create custom user application front-ends to underlying Alchemy functionality.

Customers who are familiar with applications such as MS Access, Crystal Reports or Microsoft Word can access Alchemy databases through the use of SQL query tools such as MS Query. This requires a pre-requisite understand of the SQL query tool (i.e. MS Query) including the ability to select and connect to ODBC data sources, and the ability to formulate and execute structured queries using graphical tools.

Customers who are interested in integrating Alchemy functionality into custom applications must have pre-requisite knowledge of the ODBC v2.5 API function calls as well as of their chosen development tool (i.e. MS Visual Basic, MS Visual C++, Microsoft .NET languages, PowerBuilder, etc.)

## 5.3 *Sample Applications*

**Banking:** Credit and loans manager can use Crystal Reports to generate a report of all loans approved in 2002, which were between \$1 million and \$10 million. The 2002 loan information, archived to an Alchemy database would be accessed by Crystal Reports which would then perform the calculations necessary to generate the desired report.

**Medical/Healthcare:** Marketing employees of an HMO could use Microsoft Word's mail merge capability and Microsoft Query to query an Alchemy patient database to create a mailing list for all cancer patients to inform them of a new treatment now available

**Legal:** The MIS department of a large legal firm could develop a custom internal application using Visual Basic. This application would provide lawyers the ability to search their own current client files, stored on their hard drives, as well as searching on archived legal documents stored in Alchemy databases, all through a common and customized user interface.

## 5.4 ODBC functionality not implemented in Alchemy 7.4

1. Compound predicates involving a regular predicate ORed with the extended predicate. While compound predicates are supported in the general case, e.g. `SELECT * FROM xx WHERE ParentId = 1257 AND {PRED Full_Text, CONTAINS, 'paid'}` the CONTAINS clause cannot be ORed with a regular predicate.

`SELECT * FROM xx WHERE ParentId = 1257 OR {PRED Full_Text, CONTAINS, 'approved'}`  
Note that this limitation does not apply to CONTAINS predicates ORed together:

`SELECT * FROM xx WHERE {PRED Full_Text, CONTAINS, 'paid'} OR {PRED Full_Text, CONTAINS, 'approved'}`

2. Nested SELECT statements using IN:  
Statements of the following type are not supported:

`SELECT * FROM xx WHERE Id IN (SELECT ParentId WHERE {PRED Full_Text ONTAINS, 'approved'})`

3. Prepared statement involving compound extended predicates:  
`SELECT * FROM xx WHERE {PRED Full_Text, CONTAINS, ? }  
AND {PRED Full_Text, CONTAINS, ? }  
OR {PRED Full_Text, CONTAINS, ? }`

4. Primary, Foreign Keys, and the Id Column

As implemented, the Alchemy ODBC driver does not support user-defined primary or foreign keys.

5. Multithreading

The Alchemy ODBC driver is not thread-safe. Application programs are restricted to accessing the driver through a single thread. Web applications utilizing IIS and/or .NET cannot make use of the Alchemy ODBC interface to support multiple users.

## 6. Benefits of an ODBC approach

**One Standard Interface.** One single, portable application programming interface for data access can be leveraged across diverse platforms, allowing the same ODBC installation and administration procedures to be used for both archived documents and information stored in an RDBMS, thereby minimizing administration time. Administrators can use the familiar Microsoft ODBC installation and configuration utilities. Administrators see Alchemy as just another data source.

**Access to Multiple DBMSs.** Users can access data in more than one type of DBMS—such as Oracle, Sybase, Ingres, and Informix.

**Access to Multiple Data Storage Locations.** Users can access data in more than one data storage location (for example, more than one database) at a time from within a single application. This enables end-user applications to connect to multiple data sources making the vast volumes of corporate data readily available.

**One Standard SQL.** Application developers can use one standard Structured Query Language (SQL) as defined in the ANSI 1992 standard.

**Protected Investments.** Corporations can protect their investments in existing DBMSs and in developers' DBMS skills. With ODBC, organizations can continue to use existing diverse DBMSs through the use of ODBC-aware development tools and allows developers to use standard Microsoft ODBC tools for monitoring and debugging ODBC messages.

**Insulated Applications.** Application developers no longer need to worry about changes to underlying network and DBMS versions. Modifications to networking transports, servers, and DBMSs will not affect an ODBC application.

**Leverage Industry Leading Applications.** Enables decision support access to Alchemy databases by leveraging leading industry applications. Alchemy may be used by administrators and power users to do analysis and reporting through Microsoft Word for Windows and Query in a matter of minutes; Excel and Microsoft Query can be used to analyze Alchemy data and generate reports.

**Coexistence of Drivers.** Supports integration of Alchemy within an ODBC application environment and the coexistence of Alchemy with other ODBC database drivers.

## About IMR

IMR is the premier developer of document management software that gives every organization the power to retain and access all of their vital documents. This can include scanned documents, electronic documents, forms, enterprise reports, CAD/engineering drawings and email — the information generated by everyday business processes. IMR software is used by thousands of organizations to easily store, access and utilize these documents. Founded in 1992, IMR is a privately held corporation based in Englewood, Colorado and serves customers in 42 countries. IMR made its debut in 2001 for Software Magazine's "Software 500" list, the world's most significant providers of software and software-related services.

© Copyright 2002 Information Management Research, Inc. All rights reserved.