

DATA SHEET

MICRO FOCUS DATABASE CONNECTORS

Business Challenge

Today's business climate, more than ever, drives a focus on costs within the IT organization. Where multiple technologies are being used to address a common business need there will be unnecessary costs introduced to manage those systems, build connections between them, and deal with multiple vendors. One area where this has particularly noticeable has been in the area of data storage. IT organizations often wish to standardize on a single database technology to leverage economies of scale while reducing the variety of admin and integration expertise needed to interact with multiple systems.

While this proposition is attractive, if the IT organization has a substantial investment applications that are tied to an existing data format, the cost to move may outweigh the potential benefits. With Micro Focus Database Connectors, the need for a proprietary data format for COBOL data can be lifted while not requiring a rewrite of COBOL applications. The organization sees the benefits of reduced management overhead and increased data flexibility with very little risk since they continue to run COBOL applications without change.

Product Overview

Micro Focus Database Connectors provide a binding between standard COBOL VSAM type syntax (OPEN, CLOSE, READ, WRITE) and relational database management systems (RDBMS) without having to change COBOL applications. By moving COBOL data into a RDBMS, the IT organization can use existing skills to manage the data and exploit the database provider's tools for management functions such as backup or scaling.

Key Benefits

- Support for Oracle, IBM DB2, and Microsoft SQL Server makes it easy for existing administrators (DBAs) and database skills and assets to be used
- Access data in a RDBMS without changing COBOL code or learning SQL provides real time analytics capabilities to existing COBOL assets
- Automated data type mapping from relation database types to/from COBOL makes it easy for any DBA to understand the data from COBOL without changing any of the data subsets

Detailed Feature Overview

AUTOMATIC DATA MAPPING

Database Connectors use data dictionaries to map COBOL data items to database fields. These dictionaries are also called eXtended File Descriptors (XFDs) because they're based on standard COBOL file descriptors (FDs). To generate XFDs, "CREATExFD" directives are specified when the COBOL program is compiled. The Micro Focus compiler then generates an XFD for every data file in the program.

INSULATION FROM SQL AND DATABASE RULES

Because Database Connectors handle all translations, there is no need for development staff to know SQL or to understand the unique language requirements of the database. Database rules do not need to be enforced either. Database Connectors utilize the full power of the database engine for this. Any violations are returned to the COBOL program as I/O error conditions. If new constraints are added or changes made to the database structure, the COBOL application does not have to be re-coded or changed. Even database changes do not have to be recoded. All that is required is the use of the appropriate Database Connectors interface.

CONTROL OVER DATABASE STRUCTURE

For those who want control over how the database table is built, Database Connectors support the use of directives. Directives are special comments that are placed into an FD in the COBOL source code to guide the building of the data dictionaries, which in turn guide the building of the database table. Users can specify column names, assign columns the DATE type, map elementary items of a group together, and more. Because directives are placed into the code as comments, they do not affect the functionality of the application in a non-database environment.

PERFORMANCE

Although users do not have to modify the code to use Database Connectors, users may want to add or modify code to tune performance. For instance, a WHERE constraint might be used to specify boundaries on queries that are generated or additional SELECT criteria to shift more of the data processing onto the database server. This is often the most effective way to speed table searches. To optimize performance, a user may want to restructure data, or may want certain files to remain in Micro Focus' indexed file system. Processing speed is fastest with indexed files. Alternatively, SQL can

be embedded for batch or record-level processing. (Micro Focus offers an ESQL precompiler and connects to industry-standard ESQL precompilers as well.) We recommend the use of a database administrator for performance tuning and configuration management.

FLEXIBLE CONFIGURATION

There are several variables that can be set in the runtime configuration file to influence Database Connectors processing. Some are database-specific, but most apply to all databases. Using configuration variables, users can specify the location of data files and XFDs, column case, Julian base date and more.

OTHER KEY FEATURES

- **Native Database Usage**— all access to the database is via standard SQL via its native client. This ensures all rules and constraints are honored.
- **Advanced RDBMS features** allow standard SQL features such as VIEW to be used with the COBOL data stored in the RDBMS
- **Automatic Table Structuring** create tables automatically for the COBOL data items

Product Specifications & System Requirements

The system requirements are different for each Database Connector product. To discuss the system requirements or for additional information, please contact Micro Focus.

For additional information please visit: www.microfocus.com

Copyright © Micro Focus (IP) Limited 2009. All rights reserved. The software and information contained herein are proprietary to, and comprise valuable trade secrets of, Micro Focus (IP) Limited, which intends to preserve as trade secrets such software and information. This software is an unpublished copyright of Micro Focus and may not be used, copied, transmitted, or stored in any manner. This software and information or any other copies thereof may not be provided or otherwise made available to any other person. DSDBCN1009