NOMAD
Interface for DB2

The NOMAD Interface to DB2 provides the full functionality of NOMAD for application development and reporting to maximize your investment in DB2

- Offers broad flexibility for reporting and application development for NOMAD and DB2 databases

- Is designed for optimum efficiency with the DB2 engine

- Adds relational features not found in DB2

- Offers both static and dynamic SQL access

- Provides support for IBM's DRDA

Maximize the Power of DB2

With the NOMAD Interface for DB2, you can take full advantage of all the power in DB2 by supplementing its basic query and updating facilities with NOMAD's comprehensive fourth-generation language and relational database management capabilities. First introduced in 1986, NOMAD's interface to DB2 has been cited by industry experts as providing the tightest coupling of a fourth-generation language productivity tool with the DB2 engine. As a result, the NOMAD Interface for DB2 offers the functionality, power, and efficiency to meet the complete range of reporting and application development needs, including production-level application requirements.

Application developers quickly benefit from NOMAD's complete non-procedural language for fast prototyping, its built-in decision-support functions, and its fully integrated programming language. In addition, the product offers an array of tools and templates for quickly generating report requests, forms, menus and maintenance procedures.

The NOMAD Interface for DB2 provides you with the ability to use both dynamic and static SQL access to DB2 data. This makes it equally well-suited for highly generalized, ad hoc applications, as well as production applications where performance and program authorization are key requirements.

The NOMAD Interface for DB2 enables users to access DB2 data from both TSO and MVS batch. Additionally, when used with the NOMAD Session Manager, non-TSO access is available from CICS and VTAM.

Integration with the DB2 Engine

When used to access DB2 dynamically, the interface translates NOMAD commands into SQL requests, passing off as much work as possible to DB2. Sorting and aggregation triggered by NOMAD's LIST command for reporting are performed by DB2, with only the columns needed to fulfill the data request shipped back to NOMAD. Selective data listing can be handled efficiently through SELECT and LIST WHERE commands. Global data maintenance is handled by passing set calls for CHANGE and DELETE as a single transaction. Access rights are granted directly through DB2. Additional security can be provided with NOMAD passwords, database profiles, retrieval, and update procedures. A NOMAD /DB2 Resource Governor is also available to limit the amount of resources used by interactive users. Concurrent access by multiple users and data sharing are handled by DB2. DB2 tables can be created by NOMAD's data definition language using a single command, SCHEMA NEW.

SELECT
BUSINESS SOLUTIONS

Conversely, SCHEMGEN provides an automated facility for describing DB2 tables to NOMAD. SCHEMGEN can also be used to describe DB2 system catalog information to NOMAD. The database administrator can then easily report from catalog information, which helps to provide better control over data resources.

Relational Enhancements

NOMAD interfaces efficiently with DB2 because NOMAD itself is based on a relational model. In addition, NOMAD provides important relational features not found in DB2.

Complete Outer JOIN Support:
NOMAD's MERGE MATCHING for outer joins and EXRACT ALL MATCHING for left-hand outer joins provide this support. There is no limit on the number of tables that can be joined using NOMAD.

Extended Data Type Support:
NOMAD has additional data types for fixed and varying arrays, time-series, and text data.

Enhanced Referential Integrity:
Complete support for referential integrity was available in NOMAD more than two years before it was provided by DB2. NOMAD continues to offer additional features in this area that are still not provided in DB2, including:
- A warning option which requests user approval before a cascading action takes place.
- UPDATE options of CASCADE and NAVIT allowed in addition to DENY.

Other relational features in NOMAD include:

Null Support:
NOMAD supports null values by handling missing values in a column differently from values of zeros or blanks. Nulls are disallowed for any item by specifying NOTNAV in the Schema. Additionally, NOMAD always disallows nulls for primary keys.

Using the Interface to DB2

Accessing the Data:
With the NOMAD Interface for DB2, a database can include data stored in DB2 tables, in NOMAD's native relational database, and in external files such as VSAM and QSAM. With additional NOMAD interfaces, it can also include data in other DBMSs, including Teradata. The TYPE parameter on the MASTER statement indicates how data is stored.

DB2 tables can be maintained using all of NOMAD's navigation and maintenance commands. Data stored in multiple files can be drawn together using NOMAD's relational facilities, such as DEFINE EXTRACT and MERGE MATCHING. NOMAD reporting and file creation work exactly as if the data were stored in NOMAD. NOMAD retains its concept of position, despite the fact that DB2 does not support such a concept. Thus, you have the ability to move backward and forward through the data.

Moving the Data:
Moving data from a NOMAD database to a DB2 table is accomplished simply by adding the TYPE DB2 parameter to NOMAD's Schema, or data-definition file, and issuing the SCHEMA REORG command. There is no need to worry about dumping and reloading data. The NOMAD Interface for DB2 does the work of creating a new DB2 table for you.

Using the Data:
To use NOMAD with existing DB2 tables, you use SCHEMGEN, a menu-assisted Schema generator, which produces a basic NOMAD Schema from selected tables.

This Schema can be edited to take advantage of NOMAD's extensive data-definition language to add headings, masks, limits, member checks, defined items, passwords, and multiple data types. In addition to fixed and varying arrays and time series, NOMAD data types include TEXT, DATE, TIME, and DATETIME. Additional display formats include FORMAT, PICTURE, and NAME. After the Schema is compiled, all of NOMAD's command language is available for application development, reporting, and analysis.

NOMAD for Application Development and Reporting

Integrated 4GL and Procedural Language:
NOMAD is unique in its integration of powerful non-procedural commands with a full procedural language. This integration facilitates ad hoc reporting and database maintenance, as well as quick prototyping and application development.

Programmable Windows
NOMAD Windows provides developers with a programmable windowed environment in which to work and create windowed applications. The facility automatically manages multiple screen display areas and can be used to interactively pan through output produced by reports or commands, as well as recall prior commands. Developers have complete flexibility in application interface design. Windows can incorporate structured and freeform textual fields without restriction. Any number of windows can be active simultaneously, so users can browse or update data from multiple sources without losing data or context in other windows. Additionally, windows can pop up, cascade, overlap and change

in size or screen location as needed in an application.

■ Database Editing:

A single command, DBEDIT, enables the user to enter a full-screen environment to update any DB2 table in the database without any additional programming. This feature is especially useful if your data needs to be cleaned prior to loading into DB2.

■ Application Generation Tools:

An extensive collection of facilities is available to speed application development, including templates for creating menus and forms, tools for generating maintenance procedures without coding program statements, and a tool for quickly building and revising report requests.

■ Inquiry and Reporting:

NOMAD enables users to build sophisticated production-level reporting applications for interactive or batch execution. The highly versatile LIST command provides full flexibility for row/column and free-format reporting, and is a powerful tool for financial reporting. An optional facility is available to reduce resource usage for production reporting by producing multiple reports with a single pass through the database.

■ Single Work Environment:

All commands are issued within NOMAD's single environment. While it may be desirable to shield some users from ever having to know DB2 or SQL syntax, other users will appreciate the ability to issue SQL commands such as GRANT and REVOKE from this single environment.

Static SQL Access to DB2

For the variable inquiries often associated with ad hoc reporting and analysis, the names of tables, columns, and views to be accessed are often unknown until the request is made. NOMAD's use of dynamic SQL allows the user to specify the objects at the time of request. The entire NOMAD language is available to the user, who may not even know the data is stored in DB2 tables. These characteristics are also ideal for application development, where this flexibility is especially desirable. For applications where the referenced DB2 objects are known and can be fully described during the application development phase, as is frequently the case with production systems, static SQL access provides the needed control and efficiency. Here, static SQL queries passed to DB2 can be generated, precompiled, and bound into a DB2 application plan in advance. At run time, it is only necessary for NOMAD to execute the pre-bound SQL query. This also provides tighter control over the data, as the application user only needs authorization to execute the application plans generated by NOMAD.

To facilitate the design of applications with static SQL, NOMAD provides an automated Plan Generator. The Plan Generator is a windowed pick-and-choose environment for selecting the DB2 objects for which static access is desired and the operations to go against the data. The tool generates a NOMAD description of the objects, as well as the Assembler source code with embedded SQL. The Plan Generator also produces the JCL for the program preparation (precompile, assemble, link-edit) and for binding the application plan. Host variables are supported through the Plan Generator to provide greater flexibility, allowing the variable's value to change without having to modify the plan.

DRDA Support

The NOMAD Interface for DB2 supports IBM's Distributed Relational Database Architecture (DRDA), allowing transparent access to data across multiple DRDA-compliant DB2 subsystems. DRDA allows access to remote tables as if they were local. SQL becomes the standard language that provides compatibility across environments. The NOMAD Interface for DB2 supports two levels of DRDA: Remote and Distributed unit of work

Web Enable the Data

For the Web, Select offers powerful products that Web-enable mainframe DB2 data and its application processing. Our two-tiered mainframe Web server solution integrated with the NOMAD language provides much wider access to existing applications via the Web, yet maintains the efficiency and security of the mainframe.

Supported Environments

- ■ **z/VM and z/OS TSO**
- ■ **ISPF**
- ■ **MVS Batch (native or TSO)**
- ■ **Session Manager for NOMAD access via TSO, CICS, or VTAM**

To obtain more information, please contact **Select Business Solutions at 888-472-7347 or at www.selectbs.com.**

USA

Phone: +1.888.472.7347 (NA only)
              +1.203.383.4600
Fax: +1.888.878.5178
E-Mail: info@selectbs.com

UK

Phone: +44.20.3642.0532
E-Mail: info@ selectbs.com

S SELECT

BUSINESS SOLUTIONS