



MQSeries® Integrator for OS/390®

Installation and Configuration Guide

Version 1.1.1

GC34-5762-01

Note: Before using this information, and the product it supports, be sure to read the general information under *Notices* on page 51.

Second edition (June 2000)

This edition applies to IBM® MQSeries Integrator, Version 1.1 and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

At the back of this publication is a page titled “Sending your comments to IBM”. If you want to make comments, but the methods described are not available to you, please address them to:

IBM United Kingdom Laboratories
Information Development,
Mail Point 095,
Hursley Park,
Winchester,
Hampshire,
England,
SO21 2JN.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright New Era of Networks, Inc., 1998, 2000. All rights reserved.

© Copyright International Business Machines Corporation, 1999, 2000. All rights reserved.

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Chapter 1: Introduction	5
MQSeries Integrator Overview.....	6
NEONFormatter	6
NEONRules	7
MQSeries Integrator Rules Daemon	7
MQSeries.....	7
Product Documentation Set	8
Before You Contact Technical Support.....	8
Year 2000 Readiness Disclosure.....	10
Chapter 2: Planning Your Installation	11
Installation and Configuration Overview	11
Requirements	12
Preparing Your Site for Installation.....	12
Estimating Storage Needs	14
Chapter 3: Customizing MQSeries Integrator .17	
Performing SMP/E APPLY and SMP/E ACCEPT.....	17
Installing and Updating System Parameters	17
Customizing the SQL.....	17
Secondary Authorization Ids in DB2	18
Configure DB2.....	19
Access to DB2 Tables.....	19
MQSeries Integrator Interface to DB2	21
Creating the DB2 Schema	25
Tailoring Jobs for Your Site	27
Sample Jobs.....	29
Customizing the JCL	31
Sample Programs.....	35
Control Cards	37
Creating Sample Queues	38
Connectivity Between Windows NT and OS/390 DB2	39
Using DB2 Connect	39

Chapter 4: Installing Client Components	41
Installing the GUIs	41
Chapter 5: Verifying the Installation	43
Defining Queues	43
Defining Formats	44
Defining Rules	45
Index	49
Appendix A: Notices	51
Trademarks and Service Marks	53

Chapter 1

Introduction

This guide serves as a supplement to the IBM publication, *Program Directory for MQSeries Integrator for OS/390 (GI10-2519-01)*. It provides the following information:

- Chapter 1, *Introduction*, provides an outline of the contents of this guide, a description of MQSeries Integrator and its components, the documentation set, and technical support information.
- Chapter 2, *Planning for Installation*, provides procedures for installing MQSeries Integrator libraries, tables, and utilities.
- Chapter 3, *Customizing MQSeries Integrator*, provides procedures for customizing MQSeries Integrator for your environment.
- Chapter 4, *Installing Client Components*, provides procedures for installing the NEONFormatter GUI, the NEONRules GUI, and the Visual Tester GUI on the Windows NT client.
- Chapter 5, *Verifying the Installation*, provides procedures for verifying the installation of the MQSeries Integrator for OS/390 and its components.
- Appendix A, *Notices*, provides IBM trademark and service mark information.

MQSeries Integrator Overview

MQSeries Integrator, from IBM and New Era of Networks, Inc. (NEON), provides the flexibility and scalability that allows true application integration. MQSeries Integrator consists of four components:

- IBM MQSeries
- NEONFormatter
- NEONRules
- MQSeries Integrator Rules daemon

MQSeries Integrator is a cross-platform, guaranteed delivery, messaging middleware product designed to facilitate the synchronization, management, and distribution of information (messages) across large-scale, heterogeneous networks.

MQSeries Integrator is configurable and uses a content-based rules message evaluation, formatting, and routing paradigm. MQSeries Integrator also provides a powerful data content-based, source-target mechanism with dynamic format parsing and conversion capability.

The application program interfaces (APIs) and graphical user interfaces (GUIs) allow you to use these systems. Refer to the *Programming Reference* documents for instructions on using the APIs and the *User's Guide* for instructions on using the GUIs.

NEONFormatter

NEONFormatter translates messages from one format to another.

NEONFormatter handles multiple message format types from multiple data value sources with the ability to convert and parse messages. When a message is provided as input to Formatter, the message is parsed and data values are returned.

Message formats in the NEONFormatter database are defined through the graphical user interface (GUI). The GUI leads you through the definitions of format components, for example, tags, delimiters, and patterns, to the building of complete message definitions.

NEONRules

NEONRules lets you develop rules for managing message destination IDs, receiver locations, expected message formats, and any processes initiated upon message delivery. Creation and dispatch of multiple messages to multiple destinations from a single input message is supported.

Note:

For more in-depth descriptions of NEONFormatter and NEONRules, refer to the overviews in Chapters 3 and 4 of the *MQSeries Integrator User's Guide*.

MQSeries Integrator Rules Daemon

The MQSeries Integrator Rules daemon combines MQSeries, NEONFormatter, and NEONRules in a generic server process. The MQSeries Integrator Rules daemon processes messages from an MQSeries input queue, uses NEONFormatter to parse messages, uses NEONRules to determine what transformations to perform and where to route the messages, and then puts the output messages on MQSeries queues for delivery to applications.

MQSeries

MQSeries is a message-oriented middleware that is ideal for high-value message handling and high-volume applications because it guarantees each message is delivered only once, and it supports transactional messaging. Messages are grouped into units of work and either all or none of the messages in a unit or work are processed. MQSeries coordinates message work with other transaction work, like database updates, so data integrity is always maintained.

Product Documentation Set

The MQSeries Integrator for OS/390 documentation set includes:

- ***Installation and Configuration Guide***
- ***User's Guide***
- ***System Management Guide***
- ***Programming References***
 - ***Application Development Guide***
 - ***Programming Reference for NEONFormatter***
 - ***Programming Reference for NEONRules***
- ***Program Supplement***

Before You Contact Technical Support

If you have difficulty executing one of the MQSeries Integrator programs, analyze your environment using the following steps. Be prepared to send the listed information and files to technical support.

1. Has this program ever worked in your environment?
If so, identify what has changed.
2. Check the values specified in the SQLSVSES (DD-name SQLSVSES) file that the failing job is using to make sure it refers to an existing DB2 subsystem and an existing DB2 database within that subsystem.
3. Check the values specified in the CLIINI (DD-name DSNAOINI) file that the failing job is using to make sure it refers to an existing DB2 subsystem and an existing DB2 database within that subsystem.

4. Check whether the System Affinity is causing your job to execute on a system that does not contain the DB2 subsystem, MQSeries queue manager, or IBM datasets that MQSeries Integrator is trying to access.
5. In the CLIINI file (DD-name DSNAOINI), edit the following line:

```
CLITRACE=0
```

Change it to:

```
CLITRACE=1
```

Rerun your job. The CLITRACE produced (DD-name CLITRACE) is invaluable in diagnosing problems between the DB2 database and the MQSeries Integrator application. Your JCL should have a DD-statement that defines CLITRACE to either a disk file or SYSOUT class. This file is required by technical support to diagnose problems.

Note:

It is assumed that the DB2 CLI is installed, the DSNACLI Plan has been bound, and you are granted execute authority on it.

6. Examine all files produced by MQSeries Integrator for error or informational messages. Some error messages are written to SYSOUT, some to SYSPRINT, and some to STATLOG.
7. Look for Operating System messages that may indicate why the job has failed, such as missing files, no room to log messages (E-37, B-37 type failures), full queue conditions, and so on.
8. If failing to put or get from an MQSeries queue, make sure the queue is enabled for sharing:


```
Permit shared access . . . . Y Y=Yes,N=No
Default share option . . . . S E=Exclusive,S=Shared
```
9. If the problem is related to poor Rules daemon performance, check the values of the timers specified in the input stream (DD-name SYSIN) file of the RULENG job. Setting these timers too high can result in poor performance of the Rules Engine.

When contacting technical support be prepared to send the following information via email or ftp:

- The complete listing of your jobs execution, including SYSOUTs, SYSPRINTs, STATLOG, JESMSGs, and so forth.
- The contents of the CLITRACE file
- Any dump files produced (CEEDUMP or SYSUDUMP)
- Your site's SQLSVSES file
- Your site's CLIINI file

Year 2000 Readiness Disclosure

MQSeries Integrator, when used in accordance with its associated documentation, is capable of correctly processing, providing, and/or receiving date information within and between the twentieth and twenty-first centuries, provided that all products (for example, hardware, software, and firmware) used with this IBM program properly exchange accurate date information with it.

Customers should contact third party owners or vendors regarding the readiness status of their products.

IBM reserves the right to update the information shown here. For the latest information regarding levels of supported software, refer to:

<http://www.software.ibm.com/ts/mqseries/platforms/supported.html>

For the latest IBM statement regarding Year 2000 readiness, refer to:

<http://www.ibm.com/ibm/year2000/>

Chapter 2

Planning Your Installation

This chapter provides the procedures for installing the MQSeries Integrator libraries, tables, and utilities.

The installation process requires the following experience:

- DB2 database administration and user-security
- ODBC client connections
- Installation planning and coordination
- Supporting transaction monitors and communications
- Working with IBM MQSeries queues
- Writing JCL
- Using TSO

Installation and Configuration Overview

The installation and configuration process for MQSeries Integrator includes:

- Preparing your site
- Configuring your DB2 database
- Tailoring JCL files
- Testing MQSeries Integrator
- Installing the GUI utility

Requirements

Hardware

- An IBM compatible mainframe capable of running the software listed below.
- See *Estimating Storage Needs* on page 14 for formulas to help calculate your approximate DB2 space requirements, depending on installation and use.
- IBM-compatible PC with CD-ROM drive to run the GUI utilities and to view documentation.

Software

- Windows NT 4.0 to run the GUI utilities and to view the documentation; requires approximately 60 MB of disk space on the end user's computer
- ODBC driver to connect MQSeries Integrator GUI utilities to OS/390 and DB2

Note:

With DB2 v5.1 and OS/390 v2.5 or later, you can use either SNA (LU6.2) or TCP/IP to connect the ODBC client to DB2 on OS/390.

Preparing Your Site for Installation

Before installing MQSeries Integrator, you should know the following:

- Number and identity (userid) of concurrent users on your system.
- DBA userid for the owner (DBADM) of the MQSeries Integrator DB2 databases.
- DB2 database name for MQSeries Integrator tables.
- DB2 subsystem name where the MQSeries Integrator databases will reside.
- DB2 storage groups for MQSeries Integrator tablespaces.

- DASD volumes and catalog where you will create DB2 storage groups for MQSeries Integrator tablespaces.

You must have facilities and procedures to perform the following:

- Run DB2 SPUFI scripts to create databases, tablespaces, tables, and indices, and then grant access to them. These scripts are provided with the MQSeries Integrator product.

Estimating Storage Needs

NEONFormatter Space Requirements

To calculate the total database space required for NEONFormatter, estimate the number of objects used by NEONFormatter. Determine a number for each of the following objects. These numbers are used in the DB2 formula in the next section.

- Input parse controls
- Delimiters
- Output format controls
- Flat input formats
- Flat output formats
- Compound formats
- Fields in flat input, flat output, and compound formats

DB2

Use the following formula to determine the space required for NEONFormatter. A key to the formula expressions follows the formula.

$$\begin{aligned}
 &(1821 + (168 \times F)) + \\
 &(1821 + (56 \times D)) + \\
 &(1821 + (200 \times IPC)) + \\
 &(84 \times NV) + \\
 &(1821 + (615 \times OFC)) + \\
 &(1821 + (88 \times FIF)) + \\
 &(1821 + (100 \times FOF)) + \\
 &(1821 + (84 \times CF)) = \\
 &FSpace
 \end{aligned}$$

Key

Symbol	Description
F	fields
D	delimiters
IPC	input parse controls
NV	name/value input field validation parameter pairs
OFC	output format controls
FIF	flat input formats
FOF	flat output formats
CF	compound formats
FSpace	Total space, in bytes, required for NEONFormatter

NEONRules Space Requirements

To calculate the total database space required for NEONRules, estimate a number for each of the following components. These numbers are used in the DB2 formula in the next section.

- Application groups
- Message types within each application group
- Rules names within each message type
- Arguments within each rule name
- Actions within each subscription

DB2

Use the following formula to determine the space required for NEONRules. A key to the formula expressions follows the formula.

$$(1322 + (47 \times AG)) +$$

$$(1330 + (17 \times MT)) +$$

$$\begin{aligned}
 &(1330 + (601 \times R)) + \\
 &(1330 + (242 \times A)) + \\
 &(1330 + (260 \times S)) + \\
 &(127 \times AO) = \\
 &RSpace
 \end{aligned}$$

Key

Symbol	Description
AG	application groups
MT	message types within an application group
R	rule names within each message type
A	expressions within each rule
S	subscriptions within each rule
AO	actions within each subscription
RSpace	Total space, in bytes, needed for NEONRules

Calculate the Tablespace Requirements

You should factor in an additional 15% to the space requirements to allow for additional rules and formats. Sufficient disk space must be made available for the storage groups in which you create MQSeries Integrator tablespaces.

Chapter 3

Customizing MQSeries Integrator

This chapter describes the procedures for customizing MQSeries Integrator for your environment.

Performing SMP/E APPLY and SMP/E ACCEPT

It is not necessary to perform the SMP/E ACCEPT function immediately after the SMP/E APPLY function. You can continue installing and updating system parameters following each successful completion of the SMP/E APPLY function. However, be sure to perform the SMP/E ACCEPT function before installing any maintenance or user modifications.

Installing and Updating System Parameters

Customizing the SQL

The MQSeries Integrator product uses DB2 tables to store format and rule definitions. To create and maintain these tables, MQSeries Integrator is distributed with a library of SQL DDL statements. Most members of the <smphlq>.SNEOSQL library must be tailored to your installation and naming standards prior to execution. Edit each member and replace the <substitution

string> with a value that is valid for your site. Each substitution string is delimited by left and right chevrons (< >).

SQL statements can be processed in various ways, for example, using SPUFI, QMF, a batch program such as DSNTIAR, or third-party tools. The remainder of this installation guide assumes that you are using SPUFI to process the SQL statements.

Secondary Authorization Ids in DB2

In the RACF environment, RACF groups are created as the qualifiers of DB2 databases, for example, NN41DEV1, NN41QA2, and so on. The benefit of this is that no individual user owns the database. The added advantage is maintaining ownership and maintenance of the databases in a separate group.

The RACF administrator must connect the individual TSO userids to the RACF groups or databases to which the user should have access. The RACF administrator can view the RACF groups as databases because of the one-to-one correspondence between RACF group names and the databases created with a group name as table qualifier.

RACF Settings

The following RACF settings must be enabled for this connection to work:

1. Group Access (GRPACC) must be enabled for each user ID.
2. List of Groups option must be enabled in RACF to allow all the attached RACF groups to be considered as a list of valid Auth Ids in DB2.
3. RACF must be refreshed or the system IPL'ed for this to take effect.

Secondary Auth Id Exit

The secondary Auth Id exit (the DSN3SATH and DSN3SSGN members of the DB2 SDSNSAMP library) must be assembled into the DB2 SDSNEXIT library. Installation of this exit is optional during DB2 Installation, but the exit is required for secondary Auth Ids to function. After this exit is installed using the JCL provided by IBM, DB2 must be cycled.

Configure DB2

Prior to installing MQSeries Integrator for OS/390, security, access, and connectivity requirements of MQSeries Integrator must be discussed with the customer's database administrator (DBA) and security department. The customer must provide the necessary facilities to support security, access, and connectivity.

MQSeries Integrator has two major components: the graphical user interface (GUI) programs that are installed on a Windows NT workstation and the engine code and utilities that are executed on OS/390. The GUIs are used to input rules and formats into DB2 databases. The engine code and utilities read this data from the DB2 database while executing.

Access to DB2 Tables

When you create MQSeries Integrator DB2 tables, you must consider table ownership and how users access to tables. The MQSeries Integrator product uses unqualified table names. This means that when statements such as "SELECT * FROM NNF_FMT" are executed by MQSeries Integrator, they rely on DB2 internal mechanisms to direct the query to the correct table. The default behavior in DB2 is that the user's TSO userid is prefixed to the table name. Unless the userid is the one that created the table, this results in a "Table Not Found" condition.

The following methods, in descending order of popularity, are available to solve this problem:

1. Secondary Authorization Ids

The userid accessing the table must have a valid secondary Auth Id. Create a RACF group and use the RACF group as the qualifier or SQLID when the MQSeries Integrator tables are created. The userid is then attached to the RACF group, so the RACF group becomes a secondary Auth Id for the user's userid.

You must install and use a DB2 secondary Auth Id user exit found in: '<db2hlq>.SDSNSAMP(DSN3SATH)'. The RACF List of Groups option in RACF must be active.

To test whether you have sufficient access authority, use the IBM products SPUFI or QMF to process the following SQL statements:

```
SET CURRENT SQLID = '<table qualifier>';
SELECT * FROM NNF_FMT;
```

Note:

The test SQL statements should be entered using a userid that does not have SYSADM authority.

If both SQL statements succeed, then you have access to the MQSeries Integrator tables.

For a more complete discussion of secondary Auth Ids, see the **IBM DB2 Administration Guide** (SC26-8957).

2. Synonyms

You can create synonyms for each user who must access the tables. To the user, the synonym looks like a table with the userid as the qualifier. During SQL statement processing, DB2 redirects the SQL to the real table. The Create Synonym SQL statement is similar to:

```
SET CURRENT SQLID = '<userid>';
CREATE SYNONYM NNF_FMT FOR <real-qualifier>.NNF_FMT;
```

DB2 treats the reference to <userid>.NNF_FMT as an existing table, and redirects the query to <real-qualifier>.NNF_FMT. In this case, the MQSeries Integrator GUI qualifier is the same as the userid.

One of the disadvantages in using synonyms is that a synonym must be created for every user who accesses every MQSeries Integrator table. If a table is dropped, all the synonyms to that table are also automatically dropped. The MQSeries Integrator product distribution SQL library contains a CRSYNS script to create synonyms for MQSeries Integrator tables. Using this script saves the DBA time when creating synonyms.

3. SYSADM authority

The **IBM DB2 Administration Guide** states:

An ID with SYSADM authority can set the current SQL ID to any string of up to 8 bytes, whether or not it is an authorization ID or associated with the process that is running.

Note:

Granting SYSADM authority to all MQSeries Integrator users is strongly discouraged.

MQSeries Integrator Interface to DB2

MQSeries Integrator uses the DB2 Call Level Interface (CLI) that was introduced with DB2 v5.1. The installation of the Call Level Interface is not part of the standard DB2 installation. It is documented in ***DB2 for OS/390 Version 5 Call Level Interface Guide and Reference*** (SC26-8959). If you have not installed and configured the CLI in your environment, you should not proceed any further with the MQSeries Integrator installation until the DB2 CLI is installed and running.

The MQSeries Integrator JCL library includes a member called BINDCLI to help you bind the IBM DBRMs and create a plan called DSNACLI. This JCL was adapted from the JCL shipped by IBM in DSN510.SDSNSAMP member DSNTIJCL.

Call Level Interface

The Call Level Interface (CLI) is a series of APIs that correspond to the ODBC standard and allow the porting of applications written for ODBC-compliant databases to DB2 on OS/390. The CLI API calls are documented in the same IBM manual as the CLI installation. APIs such as SQLConnect, SQLPrepare, and SQLAllocStmnt are used by MQSeries Integrator. These API calls are resolved at run time using a DLL that is part of the DB2 load library, DSN510.SDSNLOAD member DSNCLI. For example, when MQSeries Integrator executes the SQLConnect API call, the DSNCLI DLL is fetched into memory and performs a Call-Attach type of connection to DB2. All subsequent SQL statements flow through this connection between DB2 and the MQSeries Integrator application.

There are several implications of this type of functionality:

- The SQL processed by the CLI (between MQSeries Integrator and DB2) is all Dynamic SQL. There is no static or embedded SQL in the MQSeries Integrator application, and therefore, no DBRMs that are part of the MQSeries Integrator installation.

- Connection and other parameters are provided by two separate input parameter files. The first is the SQLSVSES file in the <hlq>.SNEOCNTL library. Each line of the SQLSVSES file looks like the following example.

```
SESSION_TAG:<ddf-location>:N/A:N/A:<sqlid>:
new_format_demo:DSN1:xxxxx:yyyyy:NN41QA1:
```

The parameters used in this example are described in the following table.

Parameter	Description
new_format_demo	session name used in the OpenDbmsSession API call
DSN1	DB2 DDF location name for use in the SQLConnect API call (in this example)
xxxxx	user-id; not used for security reasons
yyyyy	password; not used for security reasons
NN41QA1	qualifier for MQSeries Integrator tables in this example. For use in the "Set Current SQLID=" statement.

The value specified in the sqlid field is used in the following SQL statement as soon as the SQLConnect completes:

```
SET CURRENT SQLID = '<sqlid>';
```

The second parameter file that the CLI uses is an initialization or INI-type file that contains additional parameters used by the CLI. This file is member CLIINI and is in the <hlq>.SNEOCNTL library. It is allocated to the MQSeries Integrator process via the DD-name DSNAOINI in the JCL.

The parameters in this file are discussed in the IBM *Call Level Interface Guide and Reference* manual (SC26-8959).

```

; COMMON parameters to connect to DB2 subsystem
[COMMON]
MVSDEFAULTSSID=<db2-ssid>
TRACE=0
CLITRACE=<0=no trace; 1=trace>
TRACEFILENAME=DD:CLITRACE

; Subsystem-specific parameters
[<db2-ssid>]
MVSATTACHTYPE=CAF
PLANNAME=DSNACLI

; Data-Source specific parameters
[<database-name>]
CURRENTSQLID=<auth-id>
AUTOCOMMIT=0
OPTIMIZEFORNROWS=100
DBNAME=SYSIBM
SCHEMALIST="CURRENT SQLID, 'SYSIBM' "
TABLETYPE=" 'TABLE' , 'SYSTEM TABLE' "
UNDERSCORE=0

```

- Performance tuning DB2 for dynamic SQL is different from tuning for embedded static SQL. The DBA must execute the RUNSTATS DB2 utility on a regular basis against the MQSeries Integrator database. The RUNSTATS member of the <hlq>.SNEOJCL and STATS member of the <hlq>.SNECNTL libraries are provided to facilitate the collection of DB2 statistics for MQSeries Integrator tables. Current DB2 statistics help the DB2 optimizer select an optimal access path for the data. Other DB2 performance and tuning guidelines are outside the scope of this document.

Running SPUFI Script Files

Prior to running the scripts provided by NEON, you must review each DB2 SPUFI script file and substitute values for specific strings used as placeholders in the SQL DDL. These substitution tokens are embedded in the SQL DDL files. Before executing any of these DDL files (either in SPUFI or Batch), you must EDIT each file and replace any substitution tokens with values that are correct for your particular installation.

The following table describes the substitution tokens and provides a column to record your installation's values:

Substitution Token	Description	Installation Values
<auth-id>	The SQLID of the owner of the MQSeries Integrator database. This SQLID must have sufficient privileges on the MQSeries Integrator database to create and load the MQSeries Integrator tables. GRANTS and SYNONYMS scripts are provided to allow others to access the MQSeries Integrator database tables. This can be an owning user or an RACF group. Secondary auth-id is recommended.	
<database-name>	The name of the MQSeries Integrator database in which all tablespaces are created. This is a logical entity in the DB2 catalog used to group tables, views, indexes, and so on. The database name can be any value, but names such as NEO41DEV, NEO41QA, or NN41PROD are strongly recommended to minimize conflicts with future MQSeries Integrator releases.	
<icf-catalog-name>	Identifies the Integrated Catalog Facility catalog HLQ for the tablespaces being defined.	
<storage-group-name>	The name of the DB2 Storage Group in which the database and tablespaces are created. This defines a logical entity in the DB2 catalog that is used when allocating physical datasets for DB2 tablespaces.	

Substitution Token	Description	Installation Values
<vol-1>	Defines a volume for the storage group. Each volume-id is a volume serial number of a storage volume. It can be a maximum of 6 characters and is specified as an identifier or a string constant. vol-1, vol-2, and vol-3 are used in the CRSTOGRP (Create Storage Group) SPUFI script file.	
<vol-2>	See <vol-1>	
<vol-3>	See <vol-1>	
<synonym-owner-auth-id>	Qualifier for the created tables, if creating SYNONYMS is required.	

If there are execution errors in SPUFI or in Batch after modifying the SQL DDL files, see the IBM *DATABASE 2 for MVS/ESA SQL Reference* (SC26-8966) to diagnose the syntax error.

Note:

Many of the tables have columns that are used for filler to force rows to take up an entire DB2 page. Do not modify the schema. MQSeries Integrator utilities and programs depend on the existence of these columns.

Creating the DB2 Schema

It is assumed that you will use the IBM SPUFI utility to execute the necessary SQL scripts to create and load the MQSeries Integrator database. These files are found in the <hlq>.SNEOSQL library.

The authorization ID that you use should have DBADM authority over the MQSeries Integrator database. In some installations, this may be a two-step process as in the following example:

1. DBA-1 creates the MQSeries Integrator database and grants DBADM authority on the MQSeries Integrator database to DBA-2.

2. DBA-2 creates the MQSeries Integrator tablespaces, tables, and indexes and loads the metadata.

Each script should be edited, replacing the substitution tokens with values that are valid for your installation, and executed in the following order:

Step	Member	Description
1	CRSTOGRP	Create the MQSeries Integrator storage groups where the tablespaces will be stored.
2	CRDB	Create the MQSeries Integrator database.
3	CRTSPACE	Create MQSeries Integrator tablespaces. An alternate script CRTSPAC1 is provided to create one large tablespace.
4	CRTABS	Create MQSeries Integrator tables and indexes. An alternate script CRTABS1 is provided to create the MQSeries Integrator tables in the single tablespace created using CRTSPAC1.
5	META	Load metadata into MQSeries Integrator tables.
6	GRANTS	Grant access to MQSeries Integrator tables. The default is PUBLIC.
7	CRSYNS	(Optional) Create synonyms for MQSeries Integrator tables.

Correcting Mistakes

There are several SPUFI scripts that are useful for correcting mistakes or quickly cleaning out the database after acceptance testing. These script files are in the <hlq>.SNEOSQL library.

SPUFI Script	Task Performed
REVOKE	Revokes permissions previously granted to access the MQSeries Integrator tables.

SPUFI Script	Task Performed
TRUNCATE	Deletes all data from all MQSeries Integrator tables; this is sometimes called truncating the tables.
DRSYNS	Drops a set of synonyms pointing to MQSeries Integrator tables.
DRTABS	Drops all MQSeries Integrator tables.
DRTSPACE	Drops all MQSeries Integrator tablespaces.
DRDB	Drops the MQSeries Integrator database and all subordinate objects.
DRSTOGRP	Drops the MQSeries Integrator storage groups.
RESETDB	Returns the database tables to their state immediately following installation, for example, empty except for MQSeries Integrator metadata. This script is a combination of TRUNCATE and META.

Tailoring Jobs for Your Site

JCL that contains a statement enclosed by chevrons

(< >) means that the user must provide a valid value in that statement prior to submitting the job. For example, the APITEST job contains the following line:

```
//MSGIN DD DSN=<your-message-file>,DISP=SHR
```

The user must provide an MVS dataset name for the file containing the message data.

Each job uses in-stream procedures that contain symbolic parameters. These symbolic parameters might have to be tailored from the default installation values to match dataset names and dataset high-level qualifiers (HLQs) for your site. Each job uses some combination of the following parameters:

Parameter	Description	Default Value
PRM=(')	Runtime parameters passed to the program at startup.	varies
CEEHLQ	HLQ for IBM Language Environment run-time libraries.	
CSSHLQ	HLQ for IBM Callable System Services (CSSLIB) library	SYS1
SQLMEM	The member of the <hlq>.SNEOCNTL library containing control cards for DB2 access.	SQLSVSES
OPCLAS	Output class for SYSOUT statement.	*
INIMEM	Controls access to DB2.	CLIINI
MPF=	The member of the <hlq>.SNEOMPF library containing control cards for MQSeries access.	PUTDATA, RULENGP, or GETDATA
TCPHLQ	HLQ for TCP connection; member of SNEOPROC library.	TCPIP
LEHLQ	HLQ for current version of language environment used by the compiler; member of SNEOPROC library.	

Sample Jobs

The <hlq>.SNEOJCL and <hlq>.SNEOPROC libraries contain members that perform the following functions:

Contents of the Library

Member	Description
APITEST	Runs the APITEST program.
APITESTC (optional)	Compiles and links the MQSeries Integrator program APITEST.
BATCHDDL	Runs SQLDDL in batch if DB2 program DSNTIAD is installed.
BINDCLI	JCL and control statements used to bind DB2 CLI (if this has not been done at your site).
BINDUSRX	Binds user exit DBRMs with IBM DB2 CLI DBRMS into a new plan.
GETDATA	Runs the MQSeries Integrator program MQIGETDA.
JOBCARD	Substitute with the valid JOBCARD for your site.
MQIGETDC (optional)	Compiles and links MQIGETDA.
MQIPUTDC (optional)	Compiles and links MQIPUTDA.
MQIRULEC (optional)	Compiles and links MQIRULEN.
MQSCRIPT	Creates MQSeries queues using QDEFS input file and runs the CSQUTIL utility to create and modify MQSeries queues in batch.
MSGTEST	Runs the MQSeries Integrator program MSGTEST.
MSGTESTC (optional)	Compiles and links the MQSeries Integrator program MSGTEST.
NEOMQCC	Queue consistency checker job.

Member	Description
NNFIE	Runs the MQSeries Integrator program NNFIE.
NNRIE	Runs the MQSeries Integrator program NNRIE.
NNRTRACE	Runs the MQSeries Integrator program NNRTRACE.
OPTNOOE	Compile options file.
PUTDATA	Runs the MQSeries Integrator program MQIPUTDA.
RENAME	Renames components in an import file.
RULENG	Runs the MQSeries Integrator Rules daemon.
RULETEST	Runs the MQSeries Integrator program RULETEST.
RULESTC (optional)	Compiles the MQSeries Integrator program RULETEST.
RULOWNER	Runs the MQSeries Integrator program RULOWNER.
RUNSTATS	Perform DB2 RUNSTATS against the MQSeries Integrator database. Note: Because of the dynamic nature of the DB2 CLI, frequent use of RUNSTATS against the MQSeries Integrator database is recommended.
STEPLIB	Modify with valid dataset names for your site.
UETEST	Runs the MQSeries Integrator program UETEST.
UETESTC	Compiles and links the MQSeries Integrator program UETEST.
USEREXIT	Compiles the NNUSER DLL module used for User Exits.
USREXIT2	Compiles the NNUSER DLL module used for User Exits that contain static, embedded SQL.
CBCCL15	Member in SNEOPROC library.
CBCDB2CL	Member in SNEOPROC library.

Customizing the JCL

Each member in the <hlq>.SNEOJCL and in the <hlq>.SNEOPROC library must be edited and customized prior to running any of the jobs.

1. Edit the JOBCARD member and replace the text with a valid jobcard for your site.
2. Edit the STEPLIB member and replace the high-level qualifier (HLQ) in chevrons with names that are valid for your site. You must uncomment these links after they are changed.

Symbolic Parameters	Description	Values for Your Site
<db2hlq>	HLQ for the DB2 runtime library.	
<c++hlq>	HLQ for the C/CH compiler runtime library.	

Note:

Symbolic parameters are used in the STEPLIB definition. These symbolic parameters are defined in the instream PROCs defined in each job of the <hlq>.SNEOJCL library.

```
//STEPLIB DD DISP=SHR,DSN=&SMPHLQ..SNEOULOD      MQI User-Exit LOAD
//          DD DISP=SHR,DSN=&SMPHLQ..SNEODLL      MQI DLL Library
//          DD DISP=SHR,DSN=&SMPHLQ..SNEOLOAD     MQI Executables
//          DD DISP=SHR,DSN=&MQSHLQ..SCSQANLE     MQS Natl Lang Supt
//          DD DISP=SHR,DSN=&MQSHLQ..SCSQLOAD     MQS runtime
//          DD DISP=SHR,DSN=&MQSHLQ..SCSQAUTH     MQS runtime
//          DD DISP=SHR,DSN=&CEEHLQ..SCEERUN      Lang Envir runtime
//          DD DISP=SHR,DSN=&CSSHLQ..CSSLIB       System CSSLIB
// *
// * NOTE: The following libraries should only be used in the STEPLIB
// *       if they are not present in your system's LINKLIST.
// *
//          DD DISP=SHR,DSN=<db2hlq>.SDSNEXIT     DB2 runtime exits
//          DD DISP=SHR,DSN=<db2hlq>.SDSNLOAD     DB2 runtime
```

```

/** Choose one of the following, if needed
/** EITHER DD DISP=SHR,DSN=<c++hlq>.SCLBDLL      v 2.4 C++ runtime
/** OR      DD DISP=SHR,DSN=<c++hlq>.SCLB3DLL     earlier C++ runtime
/**

```

The preceding STEPLIB definition shows the datasets required at run time by MQSeries Integrator programs. If any of these datasets are in your site's Link Pack Area or Linklist, they can be deleted from the STEPLIB definition. However, you might have to delete the associated symbolic parameter from the instream PROCs in each JCL member.

Note:

Depending on the compiler release level, <c++hlq>.SCLBDLL or <c++hlq>.SCLB3DLL must be in your site's Linklist, or it must be added to the STEPLIB of each job. This library contains C++ runtime functions that are required at run time for the jobs to function properly.

3. Edit each member of the <hlq>.SNEOJCL library. Copy the jobcard for your site, replacing the following text:

```
<insert a valid jobcard for your site>
```

You can use the COPY command from the ISPF editor.

4. Copy the STEPLIB member, replacing the following text:

```
<tailor the member STEPLIB for your site and copy it
here>
```

You can use the COPY command from the ISPF editor.

5. Search for other substitution strings that should be tailored prior to running the job. Substitution strings are delimited by left and right chevrons (< >). For example, you must modify the following line prior to submitting this job:

```
//MSGIN      DD DSN=<your-message-file>,DISP=SHR
```

The strings to replace depend on the function or the individual test job.

6. Change the default values for the symbolic parameters to the high-level qualifiers (HLQ) for the datasets at your site.

BINDCLI

The BINDCLI JCL is provided for use if your site does not have the DB2 CLI installed. The CLI must be installed and working prior to running any MQI jobs. Make sure you are granted the necessary permissions to execute the CLI plan.

```

/** <insert a valid jobcard here >
/**
/** THIS JCL IS DISTRIBUTED FOR DB2 v5.x CUSTOMERS WHO MAY NOT
/** HAVE INSTALLED THE DB2 CALL LEVEL INTERFACE (CLI) PLANS
/**
/**
/** NOTE: Installation and use of the DB2 v5 Call Level
/**       Interface is documented in IBM manual number SC26-8959.
/**
/*******
/** JOB NAME = DSNTIJCL                               */
/**                                               */
/** DESCRIPTIVE NAME = INSTALLATION JOB STREAM      */
/**                                               */
/**       LICENSED MATERIALS - PROPERTY OF IBM      */
/**       5655-DB2                                   */
/**       (C) COPYRIGHT 1982, 1997 IBM CORP.  ALL RIGHTS RESERVED.
/**                                               */
/**       STATUS = VERSION 5                        */
/**                                               */
/** FUNCTION = SAMPLE CLI BIND                      */
/**                                               */
/** PSEUDOCODE =                                     */
/**       BINDCLI STEP BIND CLI DEFAULT PACKAGES AND PLAN
/**                                               */
/** DEPENDENCIES =                                   */
/**       CLI MUST BE INSTALLED                     */
/**                                               */
/** MEMBER DSNCLIQR CAN ONLY BE BOUND SUCCESSFULLY TO DRDA SERVERS
/** THAT SUPPORT QUERY RESULT SET SQL (I.E. DESCRIBE PROCEDURE).
/** CURRENTLY THAT IS DB2 FOR OS/390 V5.
/**

```

```

/** NOTES = */
/** BEFORE RUNNING THIS JOB: */
/** - CHANGE ALL OCCURRENCES OF <DB2HLQ> TO THE PREFIX OF YOUR */
/** DB2 V5.1 SDSNLOAD AND SDSNDBRM DATASETS */
/** - CHANGE THE SYSTEM(DSN) STATEMENT TO MATCH YOUR DB2 V5.1 SSID */
/** */
/** CLI CAN BE BOUND TO REMOTE SERVERS BY INCLUDING THE LOCATION */
/** NAME. */
/** */
/** FOR REMOTE SERVERS OTHER THAN DB2 FOR OS/390, ALSO ADD THE */
/** APPROPRIATE BIND PACKAGE MEMBER STATEMENTS, LISTED BELOW, */
/** BASED ON THE SERVER TYPE: */
/** BIND PACKAGE (<COMMON SERVER V1 LOCATION NAME>.DSNAOCLI) - */
/** MEMBER(DSNCLIV1) */
/** BIND PACKAGE (<COMMON SERVER V2 LOCATION NAME>.DSNAOCLI) - */
/** MEMBER(DSNCLIV2) */
/** BIND PACKAGE (<AS400 LOCATION NAME>.DSNAOCLI) - */
/** MEMBER(DSNCLIAS) */
/** BIND PACKAGE (<SQLDS LOCATION NAME>.DSNAOCLI) - */
/** MEMBER(DSNCLIVM) */
/** ALSO INCLUDE ANY ADDED PACKAGE NAMES TO THE PKLIST KEYWORD OF */
/** BIND PLAN STATEMENT FOLLOWING THE BIND PACKAGE STATEMENTS. */
/** */
/** ***** */
/**JOBLIB DD DISP=SHR,DSN=<db2hlq>.SDSNLOAD
/**
/**BINDCLI EXEC PGM=IKJEFT01,DYNAMNBR=20
/**DBRMLIB DD DISP=SHR,DSN=<db2hlq>.SDSNDBRM
/**SYSTSPRT DD SYSOUT=*
/**SYSPRINT DD SYSOUT=*
/**SYSUDUMP DD SYSOUT=*
/**SYSTSIN DD *
DSN SYSTEM(<db2-ssid>)

BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLICS) ISOLATION(CS)
BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLINC) ISOLATION(NC)
BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLIRR) ISOLATION(RR)
BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLIRS) ISOLATION(RS)
BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLIUR) ISOLATION(UR)
BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLIC1)
BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLIC2)
BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLIF4)

```

```

BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLIMS)
BIND PACKAGE (DSNAOCLI) MEMBER(DSNCLIQR)

BIND PLAN(DSNACLI)           -
PKLIST(DSNAOCLI.DSNCLICS)   -
    DSNAOCLI.DSNCLINC       -
    DSNAOCLI.DSNCLIRR       -
    DSNAOCLI.DSNCLIRS       -
    DSNAOCLI.DSNCLIUR       -
    DSNAOCLI.DSNCLIC1       -
    DSNAOCLI.DSNCLIC2       -
    DSNAOCLI.DSNCLIF4       -
    DSNAOCLI.DSNCLIMS       -
    DSNAOCLI.DSNCLIQR       )

END
/*

```

The function of each batch job used to support and test MQSeries Integrator is documented in *System Management*.

Sample Programs

The <hlq>.SNEOCPPLIB library contains sample programs written in C/C++ to illustrate how to use various MQSeries Integrator features. The modules and their functions are documented in the *System Management* guide. Any requisite header files are distributed in <hlq>.SNEOH.

The members of the <hlq>.SNEOCPPLIB library are:

- APITEST
- GETVAL
- MQIGETDA
- MQIPUTDA
- MQIRULEN
- MSGTEST
- NNUESTUB
- OPENFORM

- RULES
- RULETEST
- UETEST
- UETEST (in <hlq>.SNEOSRCE)

The JCL jobs to run these programs are in the <hlq>.SNEOJCL library.

Note:

The<hlq>.SNEOSRCE library is reserved for sample programs that are limited to 80-byte record lengths, for example, COBOL and C++ modules containing embedded SQL. C and C++ modules with record lengths up to 255 bytes should be placed in the SNEOCPP library.

Control Cards

The <hlq>.SNEOCNTL library contains parameter files used by various MQSeries Integrator modules at run time. The members contain <substitution-strings> that must be modified prior to executing batch jobs.

Contents of the Library

Member	Function
CLIINI	This member contains control parameters for the IBM Call Level Interface to DB2. The values in this file are documented in the IBM <i>Call Level Interface Guide and Reference</i> (SC26-8959).
QDEFS	This file contains control statements used by the MQSeries CSQUTIL utility to create sample queues that are used by the NEONRules daemon.
SQLSVSES	This member contains control statements used by the OpenDBMSession() API call, which is used by programs accessing MQSeries Integrator. The format of this file is discussed in the <i>System Management</i> and <i>Application Development</i> guides.
STATS	DB2 run stats control statements used with the RUNSTATS job. Because of the dynamic nature of the DB2 CLI, frequent use of RUNSTATS against the MQSeries Integrator database is recommended.

Creating Sample Queues

The QDEFS member of the <hlq>.SNEOCNTL library contains statements for the MQSeries utility CSQUTIL that are used to create sample queues. Replace the following substitution tokens with valid queue names for your site.

Substitution Token	Description	Values for Your Site
<your-rulesengine-input-here>	Specifies the name for the Rules daemon primary message input queue.	
<your-failure-queue>	Specifies the name of the queue where messages that fail processing within a rule are placed.	
<your-no-hit-queue>	Specifies the name of the queue where messages that do not passed Rules testing are placed.	

You must add control statements to define additional application-specific queues that might be used in PUTQUEUE actions in subscription actions.

After you edit the QDEFS files, you should tailor and run the MQSCRIPT member of the <hlq>.SNEOJCL library. This job will run the MQSeries CSQUTIL program using the edited QDEFS file as input. When this job successfully completes, your queues are defined in your queue manager.

Connectivity Between Windows NT and OS/390 DB2

The MQSeries Integrator graphical user interface (GUI) programs are WIN32-compatible PowerBuilder applications that run on a Windows NT workstation. These programs use an ODBC client connection to OS/390 to access data in DB2 tables. After the MQSeries Integrator tables are created in DB2, the ODBC client should be tested to verify that the workstation user has access to these tables.

Two ODBC clients have been tested and certified with MQSeries Integrator: IBM DB2-Connect and NEON System Shadow Direct. Installing and configuring these components are outside the scope of this document. However, ODBC-type access to DB2 tables must be installed and working prior to installing the MQSeries Integrator GUIs.

Using DB2 Connect

To verify operation of DB2 Connect on the Windows NT client PC, use the DB2 Command Line Processor in the DB2 program group and issue the following commands:

```
db2 => connect to <odbc-data-source> user <userid> using <password>
      Database Connection Information
      Database product           = DB2 OS/390 5.1.1
      SQL authorization ID      = <userid>
      Local database alias     = <odbc-data-source>

db2 => set current sqld = '<qualifier>';<press enter>

db2 => Select * from NNF_FMT;<press enter>
FMT_ID      INPUT_IND      COMPOUND_IND  FMT_NAME
-----
           1              1              0      Format1
           2              0              0      Format2

      2 record(s) selected.
db2 => quit
```

Chapter 4

Installing Client Components

This chapter describes the procedures for installing the `NEONFormatter` GUI, the `NEONRules` GUI, and the Visual Tester GUI on the Windows NT client.

Installing the GUIs

To install the GUIs on the Windows NT client:

1. Insert the MQSeries Integrator client CD-ROM into the CD-ROM drive of your Windows NT machine. If your Windows NT machine is configured to autorun from the CD-ROM, the InstallShield Wizard automatically starts.

If the InstallShield Wizard does not automatically start, choose Start→Run from the Windows Start menu. In the Run dialog box, type `setup.exe`.

2. Type or select the appropriate responses, choosing Next to continue from screen to screen.
3. On the Setup Type screen, choose one of the installation options described in the following table:

NT Client Installation

NT Options	Description of Installed Components
Typical	NEONFormatter, NEONRules, and Visual Tester GUIs MQSeries Integrator and MQSeries online documentation Readme file
Compact	NEONFormatter, NEONRules and Visual Tester GUIs Readme file

NT Options	Description of Installed Components
Custom	Only the MQSeries Integrator software components that you select are installed. A Custom Installation screen allows you to select the components.

4. Select the database and queuing to be used by Visual Tester. The database is DB2. The queuing choices are: IBM MQSeries (local queue manager), IBM MQSeries (remote queue manager), or no queuing.

Note:

If you specify local queueing or remote queueing, you must ensure that the necessary MQSeries resources are available before you use Visual Tester.

5. After the MQSeries Integrator client is successfully installed, click Finish.

Chapter 5

Verifying the Installation

This chapter describes the procedures for verifying that the components of MQSeries Integrator for OS/390 are correctly installed.

Defining Queues

1. Use the MQSeries Integrator-supplied MQSCRIPT utility and the QDEFS configuration file to define the following MQSeries local queues for testing:

Queue	Description
RulesIn	Rules daemon input queue
RulesFail	Rules daemon failure queue
RulesNoHit	Rules daemon no hit queue
Q2f	Target queue for subscription testing

2. Edit member MQSIHLQ.SNEOCNTL (QDEFS) so that it contains definitions for each queue.
3. Edit member MQSIHLQ.SNEOJCL (MQSCRIPT) to suit your environment. Submit the job so that it invokes SQUTIL to make the definitions. Check the output from the job to verify that it ran successfully. Use the MQSeries Operation and Controls panels to verify that the queues were created.

Defining Formats

1. Start the Formatter GUI.
2. Create two new literals:
 - Space, with EBCDIC value x'40'
 - Colon, with EBCDIC value x'7A'
3. Create two new fields named F1 and F2.
4. Create a new Output Operation of type Suffix named 'space' with a value of Space.
5. Create a new Output Operation Collection named 'strupspsfx' containing the individual output operations 'UPPER_CASE' and 'space'.
6. Create a new Output Control with the following properties:
 - Output Control Type: 'Data Field (Name Search)'
 - Collection Output Operation: 'strupspsfx'
 - Data Type: 'String'
 - Accept defaults for the other values.
7. Create a new Input Control named 'InCl' with the following properties:
 - Control Type: 'Data Only'
 - Type: 'String'
 - Termination: 'Delimiter'
 - Delimiter: 'Colon'
 - Accept defaults for the other values.
8. Create a new flat output format named 'Out2F'.
9. Add field components F2 and F1 (in that order) to output format 'Out2F'.

10. Apply output control OutC1 to each of the fields in output format Out2F. Right-click each field in the drop-down list of fields under the format and select **Apply Output Control**. Specify the following properties:

Field Name: Accept the default, which is the name of the field to which you are applying the control.

Output Control Name: 'OutC1'

Access Mode: 'Normal Access'

Subscript: '0'

Input Field Name: Accept the default, which is the same as the Field Name to which the control is being applied.

11. Create a new input flat format name In2F.
12. Add fields F1 and F2 (in that order) to input format In2F.
13. Apply input control InC1 to each of the fields in input format In2F. Right-click each field in the drop-down list of field under each format and select **Apply Input Control**.

The formats for testing are now complete and can be used in Rules definitions.

Defining Rules

Use the Rules GUI to define and Application Group and set of Message Types, Rules, and Subscriptions to verify Rules Processing. See the MQSeries Integrator *User's Guide* for details on how to use the Rules GUI.

Before carrying out this procedure, make sure that you have followed the procedure for defining formats. The Formats used by the Rules must be created before the Rules. Make sure that it is possible to add, modify, and delete Rules and Subscriptions. It is not possible to delete Message Types and Application Groups.

Create a Rule

1. Start the Rules GUI.
2. Create a new Application Group named AG1. Right-click **Application Groups**, click **New Application**, and type **AG1** in the text box.
3. Create a new Message Type named In2F. Click **AG1** Application Group, select the **New Message** tab in the right pane, and drag Message Type **In2F** from the Available pane to the Current pane.
4. Create a new Rule named R2F. Right-click Message Type **In2F**, click **New**, and type the name **R2F** in the text box.

Create an Expression for the Rule

1. Select the **Field List** tab in the right pane. Double-click **F1** to bring the field name to the Expression pane.
2. Select the **+/= Operators** tab and double-click **EXIST** to bring the operator to the Expression pane.
3. Select the **&sigma Functions** tab and double-click **And(&)** to bring the operator to the Expressions pane.
4. Select the **Field List** tab in the right pane. Double-click **F2** to bring the field name to the Expression pane.
5. Select the **+/= Operators** tab and double-click **EXIST** to bring the operator to the Expression pane.
6. Click the **Verify** button to verify that the expression is valid.
The Expression is OK message appears.
7. Cancel the message, and click **Apply** to apply the expression to the rule.

Create an Action List for the Rule

1. Drag **Reformat** from the Actions pane to the Action List pane.
2. Click the **INPUT_FORMAT** button.
3. Click **In2F** in the drop-down list.

4. Click in the **TARGET_FORMAT** input field, and then click the button that is displayed.
5. Click **Out2F** in the drop-down list.
6. Drag **Put Message** from the Actions pane to the Action List pane.
7. Type the name **Q2F** in the OPT_TARGET_QUEUE input field.
8. Click the **Apply** button to apply the Action List to Subscription S2F.
9. Drag Subscription **S2F** in the left pane on top of Rule R2F.

After a short time, Subscription S1F appears as an object under Rule R1F.

The hierarchy you created should have the following structure.

AG1

```

|__M In2F
    |__R R2F
        |__S S2F
            |__S Subscription List
                |__S S2F
  
```

The Rules are now complete and can be used for testing.

Index

A

accessing DB2 tables 19

B

BINDCLI JCL 33

C

C/C++

sample programs 35

calculating tablespace requirements 16

components

MQSeries 7

MQSeries Integrator Rules daemon 7

NEONFormatter 6

NEONRules 7

configuring

DB2 19

contact information for technical support 8

control card library 37

correcting mistakes 26

creating

sample queues 38

creating DB2 schema 25

customizing JCL files 31

customizing SQL statements 17

D

databases

correcting mistakes 26

DB2

accessing tables 19

Call Level Interface (CLI) 21

configuring 19

creating schema 25

secondary authorization Ids 18

DB2 Connect 39

defining

formats 44

queues 43

rules 45

documentation set 8

E

estimating storage needs 14

F

format definition 44

Formatter 6

Formatter space requirements 14

G

graphical user interfaces 41

H

hardware requirements 12

I

installation

estimating storage needs 14

overview 11

site preparation 12

installing client components

graphical user interfaces (GUIs) 41

interfacing with DB2 21

J

JCL

BINDCLI 33

control cards 37

customizing files 31

modifying 29

L

libraries

- control card 37
- SNEOCPP 35
- SNEOJCL 29

M

- mistakes 26
- modifying the JCL 29
- MQSeries Integrator
 - components 6
 - Rules daemon 7

N

- NEONFormatter 6
- NEONRules 7

P

- preparing for site installation 12

Q

- queue definition 43
- queues 38

R

- RACF settings 18
- requirements for
 - calculating tablespace 16
 - hardware 12
 - software 12
- Rules 7
- rules definition 45
- Rules space requirements 15
- running SPUFI script files 23, 27, 39

S

- sample programs
 - in C and C++ 35
- schema 25

- Secondary Auth ID Exit 18
- secondary authorization Ids with DB2 18
- SFUPI script files 27, 39
- sizing data
 - Formatter space requirements 14
 - Rules space requirements 15
- SNEOCPP library 35
- SNEOJCL library 29
- SNEOPROC library 29
- software requirements 12
- SPUFI utility 23, 25
- SQL statements 17
- SQLSVSES 37
- substitution strings 17
- substitution tokens 19
- system parameter updating 17

T

- technical support 8

U

- updating system parameters 17

Y

- year 2000 readiness 10

Appendix A

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this document to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM United Kingdom Laboratories,
Mail Point 151,
Hursley Park,
Winchester,
Hampshire,
England,
SO21 2JN.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms.

You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks and Service Marks

The following, which appear in this book or other MQSeries Integrator books, are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

MQSeries
OS/390
AIX
DB2
IBM

NEONFormatter and NEONRules are trademarks of New Era of Networks, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and/or other countries licensed exclusively through X/Open Company Limited.

Other company, product, or service names may be the trademarks or service marks of others.

Sending your comments to IBM MQSeries Integrator for OS/390, version 1.1.1 Installation and Configuration Guide GC34-5762-01

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM.

Feel free to comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness of this book. Please limit your comments to the information in this book only and the way in which the information is presented.

To request additional publications or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate, without incurring any obligation to you.

You can send your comments to IBM in any of the following ways:

- By mail, use the Readers' Comment Form
- By fax:
 - From outside the U.K., use your international access code followed by 44 1962 870229
 - From within the U.K., use 01962 870229

Electronically, use the appropriate network ID:

- IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
- IBMLink: HURSLEY(IDRCF)
- Internet: idrcf@hursley.ibm.com

Whichever you use, ensure that you include:

- The publication number and title
- The page number or topic number to which your comment applies
- Your name/address/telephone number/fax number/network ID

Readers' Comments
MQSeries Integrator for OS/390, version 1.1.1
Installation and Configuration Guide
GC34-5762-01

Use this form to tell us what you think about this manual. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer. This form is provided for comments about the information in this manual and the way it is presented.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Be sure to print your name and address below if you would like a reply.

Name	Address
Company or organization	
Telephone	Email



You can send your comments POST FREE on this form from any one of these countries:

Australia	Finland	Iceland	Netherlands	Singapore	United States
Belgium	France	Israel	New Zealand	Spain	of America
Bermuda	Germany	Italy	Norway	Sweden	
Cyprus	Greece	Luxembourg	Portugal	Switzerland	
Denmark	Hong Kong	Monaco	Republic of Ireland	United Arab Emirates	

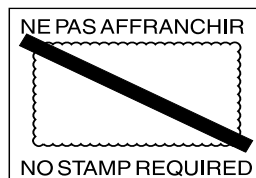
Cut along this line

If your country is not listed here, your local IBM representative will be pleased to forward your comments to us. Or you can pay the postage and send the form direct to IBM (this includes mailing in the U.K.).

2 Fold along this line

By air mail
Par avion

IBRS/CCRI NUMBER: PHQ - D/1348/SO



**REPONSE PAYEE
GRANDE-BRETAGNE**

IBM United Kingdom Laboratories
Information Development Department (MP 095)
Hursley Park
WINCHESTER, Hants
SO21 2ZZ United Kingdom

3 Fold along this line

From: Name _____
 Company or Organization _____
 Address _____

 EMAIL _____
 Telephone _____

Cut along this line

4 Fasten here with adhesive tape





Printed in U.S.A

GC34-5762-01