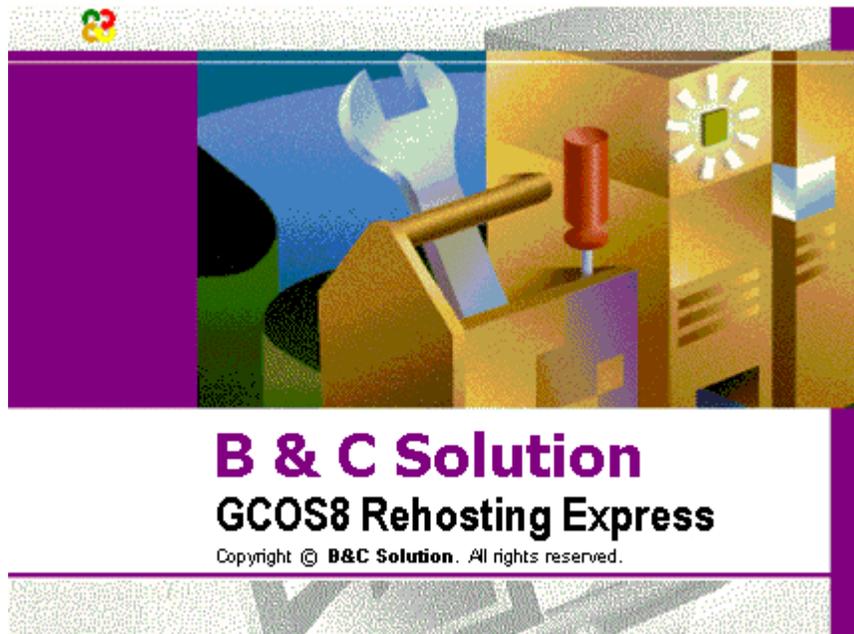


## GETTING STARTED



Getting started with GCOS8 Rehosting Express is a quick guide on how to install GCOS8 Rehosting Express, how it integrates with other deployment components and the GCOS8 Mainframe/Net Express development environment.

Version 3.1.04/4.0.04  
September 2003



# Table Of Contents

Getting started .....	1
PREFACE.....	7
Introduction .....	9
Related Documentation.....	11
Copyright information .....	12
1. RHE8 Installation.....	13
1.1 Installation information .....	13
1.2 Server Installation .....	13
1.3 Workstation Installation.....	14
1.4 Single-platform Installation .....	16
1.5 Integration with application server .....	16
1.6 Integration with development environment .....	17
2. run-time environment.....	19
2.1 running batch programs .....	19
2.2 environment setting.....	20
2.3 file mapping.....	21
2.4 running utilities .....	23
2.5 JCL/job streams .....	26
3. rdbms environment .....	29
IDS-II Techniques .....	29
3.1 IDS-II configuration.....	31
3.2 informix rdbms.....	33
Techniques .....	33
Configuration .....	33

Database setup & load .....	35
Logging.....	37
Backup .....	37
Restore .....	38
Backup - Wrapper.....	39
Restore - Wrapper .....	39
3.3 oracle RDBMS .....	40
Techniques .....	40
Configuration .....	40
Database setup & load .....	42
Logging.....	44
3.4 ibm db2 rdbms.....	45
Techniques .....	45
Configuration .....	45
Database setup & load .....	47
Logging.....	49
3.5 microsoft sql-server rdbms .....	50
3.6 interel/RFM.....	51
Configuration .....	51
Setup and Load .....	51
Application setup.....	53
3.7 DBSP .....	53
4. tpms environment .....	55
4.1 tp8 configuration .....	55
4.2 TP8 server .....	57

Principles .....	57
TP8 Server .....	57
TP8 Server Operation .....	58
TP8 Monitor .....	60
TP8 Client.....	62
4.3 bea tuxedo .....	65
Principles .....	65
Tuxedo Workstation .....	65
TP8 Configuration .....	67
TP8 Client.....	68
4.4 ibm txseries encina .....	72
4.3 micro focus enterprise server .....	73
Index .....	74



## **PREFACE**

B & C Solution has made every effort to ensure that this documentation is correct and accurate, but reserves the right to make changes without notice at its sole discretion at any time.

The Software described in this document is supplied under a license and may be used or copied only in accordance with the terms of such license, and in particular any warranty of fitness of B & C Solutions products for any particular purpose is expressly excluded and in no event will B & C Solution be liable for any consequential loss.

B & C Solution  
Ole Piisvej 4, Box 54  
DK-3100 Hornbaek  
Denmark

B & C Solution  
Grand rue, BP 1  
F-30210 Collias  
France

[www.bc-solution.com](http://www.bc-solution.com)  
[info@bc-solution.com](mailto:info@bc-solution.com)



## **INTRODUCTION**

Welcome to GCOS8 Rehosting Express, the most productive tool for deployment of your BULL GCOS8 application systems on open systems platforms.

Bull GCOS8 application systems may be maintained and tested using either GCOS8 Mainframe Express (MFE8), GCOS8 Net Express (NE8) or GCOS8 Server Express (SE8). These solutions provide a complete emulation of the GCOS8 environment.

GCOS8 Rehosting Express (RHE8) adds production features to this emulated GCOS8 environment, and thus allows you to rehost one, more or all of your GCOS8 applications systems on Windows, Linux and Unix platforms with limited effort and with excellent performance. RHE8 utilize standard software components for Database and TP, and thus provides an embedded migration of the proprietary Bull GCOS8 environment, while maintaining backward compatibility with your application system components.

### **Cobol run-time environment**

RHE8 utilize Micro Focus Application Server to provide an dynamic and efficient execution environment for your application programs. You may continue to maintain the original application programs using the MFE8, NE8 or SE8 development environment, and once compiled the programs will execute in the rehosted environment. You may also develop new COBOL application program in native Micro Focus COBOL.

### **Database environment**

Your IDS-II database(s) are rehosted on standard RDBMS software, either Oracle, IBM DB2, Informix or Microsoft SQL-Server. The migration to RDBMS is automatic, meaning once you have translated schema DDL/DMCL and subschema DDL, all the required components for definition and unload/load of your database are generated automatically. Once loaded your original application programs will access the rehosted RDBMS database(s) through standard IDS-II verbs. RHE8 utilize dynamic SQL and provides a run-time environment which may be tuned to achieve excellent performance. Database integrity rules are enforced, and you are able to develop new application with direct SQL access and use standard SQL tools to access the rehosted RDBMS database(s).

Your Interel/RFM database(s) are rehosted on IBM DB2 RDBMS.

### **TP environment**

Your TP8 Workstation(s) are rehosted on standard TPMS software, either BEA Tuxedo, IBM TXSeries Encina or Micro Focus Enterprise Server (Web Services or J2EE connector), or the embedded TP8 Workstation Monitor. RHE8 utilize Ggate from Gallagher & Robertsson to enable DSA client connections to either of the rehosted TPMS environment, allowing existing end-user implementations to be used without any changes, ex. terminal emulation Glink, VTD etc., Gweb, Web8, Gtea or Hoox.

TP8 transactions may also be accessed through client tools/interfaces provided with the rehosted TPMS environment.

### **Operational environment**

Batch JCL procedures are easily migrated to scripting and scheduling facilities on the target platform.

## **RELATED DOCUMENTATION**

GCOS8 Rehosting Express online documentation consist of:

- Getting started (this document)
- Integrated help
- Reference guide
- Add-on guides
- Release and Update notes

## **COPYRIGHT INFORMATION**

GCOS8 Rehosting Express, GCOS8 Run-time Express, GCOS8 Mainframe Express, GCOS8 Net Express and GCOS8 Server Express are trademarks of B & C Solution. All other trademarks and trade names belong to their respective companies.

© Copyright 2003 B & C Solution. All Rights Reserved.

# 1. RHE8 INSTALLATION

Before installing GCOS8 Rehosting Express you should consider whether the software is to be used on a single platform or multiple platforms. You should also consider which RDBMS and TPMS platforms you will be using for rehosting.

The software is structured to be installed into multiple directories, some directories potentially shared by multiple platforms and one directory non-shared for exclusive use.

During software installation you may select three types of installation, as follows:

- Server installation
- Workstation installation
- Single-platform installation

GCOS8 Rehosting Express is a 32-bit product and may be installed under Windows NT, 2000, XP or 2003.

GCOS8 Rehosting Express rely on Micro Focus Application Server or Enterprise Server as run-time environment, Oracle, IBM DB2, Informix or MicroSoft SQL-Server as RDBMS environment, BEA Tuxedo, IBM TXSeries or Micro Focus Enterprise Server as TPMS environment, and it is recommended to install these software components before RHE8.

## 1.1 INSTALLATION INFORMATION

The GCOS8 Rehosting Express CD use Autorun to trigger automatic installation when the CD is inserted. If automatic insert detection is disabled, you should either use 'Autorun' from popup menu or start the setup program manually from the CD.

The CD is structured as follows:

- rhe8note.txt contains Release Notes
- rhe8\disk1\setup.exe used to start installation of RHE8
- rhe8up\disk1\setup.exe used to start installation of RHE8 update

GCOS8 Rehosting Express is organized into Server and Workstation directories, and Server and Workstation installation is normally performed separately, except when RHE8 is installed in a single-platform environment.

## 1.2 SERVER INSTALLATION

Start with 'Server' installation to prepare the Server directories to be shared between multiple workstations (execution platforms). Server installation can be performed from any Workstation or on the target server (Windows NT/2000/XP/2003 only).

## **Global Directory**

The global directory holds configuration and support files.

## **Database Directory**

Database directory holds schema, subschema, backup-sets and other database support files.

## **Forms Directory**

Forms directory holds all files related to Forms.

## **Program Directory**

Program directory holds software and support files.

## **Optional Components**

- SAMPLES Files - select to install sample programs, databases etc.
- DIMS8 Files - select to install DIMS8 Forms support
- FORMAT Files - select to install FORMAT Forms support
- TSM8 Files - select to install TSM8 Forms support
- PILSPRAAK Files - select to install ArrowTalk Forms support
- SOLD+ Debeka Files - select to install SOLD+ Forms support (specific version for Debeka, Germany)
- INTEREL Files - select to install INTEREL Database support
- TEC-DIMS8 Files - select to install DIMS8 Forms support (specific version for TEC, Denmark)
- SOLD+ Barmenia Files - select to install SOLD+ Forms support (specific version for Barmenia, Germany)
- AUDAX Files - select to install QPRINT/TOOLS support (specific version for Audax, Netherlands)
- WEB8 Files - select to install WEB8 support
- DBSP - select to install DBSP support
- RHE-RDBMS - select to install Rehosting support (mandatory)

Optional components must be selected during initial installation, they cannot be added later.

## **1.3 WORKSTATION INSTALLATION**

Continue with 'Workstation' installation to prepare each Workstation (execution platform).

### **Registration**

- Name
- Company

- Registration Number

The CD cover will indicate your license registration number.

## **Local Directory**

Local directory holds configuration and work files, and must be a non-shared directory (local disk).

## **Global Directory**

Establish a link to the global directory holding configuration and support files.

## **Database Directory**

Establish a link to the database directory holding schema, subschema, backup-sets and other database support files.

## **Forms Directory**

Establish a link to the forms directory holding all files related to Forms.

## **Program Directory**

Establish a link to the program directory holding software and support files.

## **Optional Components**

Make the same selections as you did during 'Server' installation.

- SAMPLES Files - select to install sample programs, databases etc.
- DIMS8 Files - select to install DIMS8 Forms support
- FORMAT Files - select to install FORMAT Forms support
- TSM8 Files - select to install TSM8 Forms support
- PILSPRAAK Files - select to install ArrowTalk Forms support
- SOLD+ Debeka Files - select to install SOLD+ Forms support (specific version for Debeka, Germany)
- INTEREL Files - select to install INTEREL Database support
- TEC-DIMS8 Files - select to install DIMS8 Forms support (specific version for TEC, Denmark)
- SOLD+ Barmenia Files - select to install SOLD+ Forms support (specific version for Barmenia, Germany)
- AUDAX Files - select to install QPRINT/TOOLS support (specific version for Audax, Netherlands)
- WEB8 Files - select to install WEB8 support
- DBSP - select to install DBSP support
- RHE-RDBMS - select to install Rehosting support (mandatory)

Optional components must be selected during initial installation, they cannot be added later.

## **Integration**

If Micro Focus Application Server or Enterprise Server is already installed, then 'Workstation' installation will finish with an option to launch the customization process automatically. You may also do this later, see 1.5 Integration with Application Server.

If Micro Focus Application Server or Enterprise Server is not installed, then 'Workstation' installation will not offer you to launch the customization process automatically. You need to do this after installing Application Server, see 1.5 Integration with Application Server.

## **1.4 SINGLE-PLATFORM INSTALLATION**

The 'Single-Platform' installation may be used to install both 'Server' and 'Workstation' at the same time. It may be used to install GCOS8 Rehosting Express on a single-platform server, where directories are not shared with other workstations (execution platforms).

### **Single-Platform Directory**

Single-platform directory holds everything, however software components are subdivided into G, P, F, D and L subdirectories.

This technique does allow you to change to a multi-platform environment at a later stage.

## **1.5 INTEGRATION WITH APPLICATION SERVER**

Once 'Workstation' installation has been completed, you are ready to integrate GCOS8 Rehosting with Micro Focus Application Server or Enterprise Server. Besides installing files, the installation has setup information in the Windows Registry and created new shortcuts in your startup menu.

### **Registry**

Registry information, which serve as the basis for integration with Application Server, is setup under HKEY\_LOCAL\_MACHINE\SOFTWARE\B & C Solution\GCOS8 Rehosting Express\4.0\Install.

These settings should not be manually edited, however if you choose to move directories after installation, ex. going from Single-Platform to Server-Workstation, you can change these directory settings accordingly. Alternatively you can re-install Workstation install and thus change directory settings accordingly.

In either case, you need to re-run the GCOS8 Integrate Process, to reflect the changed configuration within Application Server.

## **Startup Menu**

A menu 'GCOS8 Rehosting Express 4.0' is created with the following content:

- Online Books - multiple shortcuts to various documentation
- DB Administration tools - shortcut to Interactive IDS-II and Database load utilities
- Notes - multiple shortcuts to various text documents
- Help - multiple shorts to various online help files
- Configuration - shortcut to RHE8 configuration utility
- GCOS8 Customization - shortcut to launch the customization/integration process
- RHE8 Command Prompt - shortcut to launch a command prompt for RHE8
- Run TP8 Server - optional shortcut to launch embedded TP8 Workstation Monitor

If Micro Focus Application Server was not installed before GCOS8 Rehosting Express, you may use the 'GCOS8 Customization' shortcut to launch the customization/integration process.

## **GCOS8 Customization/Integration process**

The GCOS8 customization/integration process does the following:

1. Read installation information from registry
2. Customize and create commands for batch processing under Global directory
3. Customize configuration files under Global directory
4. Customize Windows Help settings in registry (register GCOS8 help files)

The integrate process will check current settings and optional components selected, and only perform activities as required. No harm is done if the customization/integration process is run multiple times.

## **1.6 INTEGRATION WITH DEVELOPMENT ENVIRONMENT**

GCOS8 Rehosting Express is a run-time environment, which rely on either a GCOS8 Mainframe Express, GCOS8 Net Express or GCOS8 Server Express development environment for generation of application components.

The development environment must be installed with the "RHE-RDBMS" component to enable generation of application components for the RHE8 run-time environment. You need to select the same target RDBMS as for the RHE8 run-time environment.

### **IDS-II Database**

IDS-II schema(s) are translated and validated in the development environment to produce both SQL definition files and run-time support files. Use the "Move schema"

function to copy all relevant files from the development environment (G8WBDDIR) to the RHE8 run-time environment (G8RDDDIR). The SQL definition files are used to create both test and production database(s) in the target RDBMS environment.

IDS-II subschema(s) are translated and validated in the development environment to produce run-time support files. Use "Move subschema" function to copy all relevant files from the development environment (G8WBDDIR) to the RHE8 run-time environment (G8RBDDIR).

## **Interel/RFM Database**

SQL command files are migrated in the development environment, and may hereafter be used to create database(s) in the IBM DB2 target environment.

## **Configuration**

Many of the configuration options of the development environment can be moved as is to the RHE8 run-time environment, others need adaption for the run-time environment and new production related options may need to be amended to the run-time environment.

## **Application programs**

Application programs - modules, TPR's, batch programs, database procedures etc., are compiled in the development environment. Chose the GNT object format for production, so programs are executed at maximum speed. RHE8 run-time environment use dynamic linking (like the development environment), so each application components is compiled separately and need only to be made accessible to the RHE8 run-time environment.

It is recommended to copy the GNT's into one or more directories, which reside on the same server as the RHE8 run-time environment.

## 2. RUN-TIME ENVIRONMENT

The GCOS8 Rehosting Express run-time environment includes a command launch program RUN, which you use to start utilities, batch programs and TP8. The environment settings for the run-time environment is retrieved from the registry by the launch programs.

The RHE8 Command Prompt is used to start a command window with \$G8RBDIR\BIN included in the PATH settings, so launch program can be started by name, ex.

RUN command

Otherwise you must prefix the launch program with the path for \$G8RDDIR\BIN, ex.

c:\rhe8\p\bin\RUN command

Although environment settings, ex. G8RBDDIR setting, are retrieved from the registry, these settings may be overridden using the RHE8ENV.CFG file, ref. Environment setting below.

### 2.1 RUNNING BATCH PROGRAMS

You use the RUN command to launch a batch application program. The command line syntax is as follows:

```
RUN PRG ["title"] programname [cobol] [databasemode] [=output] [/localdir]
```

If programname is not given, you are prompted for command line input. Programname may be either the name of the main program or a filepath to the object file for the main program.

#### **Title**

The optional title argument may be used to assign a title to the command window, while the program is being executed. The title argument must follow PRG and must be given in quotes.

#### **Cobol**

The optional CBL85|CBL74 argument may be used to pre-load the appropriate COBOL run-time environment. The default is CBL85.

#### **Databasemode**

The optional Normal|Exclusive|Nolog|Auto|Query|NoDB argument may be used to control IDS-II database access. By default all IDS-II schemas and subschemas are pre-loaded for the COBOL run-time environment. The default is Normal. All database modes may not be supported by all target RDBMS and revert to Normal if unsupported.

Exclusive: May be used to access the database in exclusive mode (blocking all other access).

Nolog: May be used to access the database in no logging mode.

Auto: May be used to access the database with auto commit.

Query: May be used to access the database in read-only mode.

Nodb: May be used to disable pre-load of IDS-II schemas and subschemas (when application does not use IDS-II database).

## Output

The optional =filepath argument may be used to redirect the log file to the specified filepath. By default the log file is written to \$G8RBLDIR\batch.log.

## Localdir

The optional /path argument may be used to override G8RBLDIR setting with the specified path. The value /. may be used to set G8RBLDIR to current directory.

## Configuration

The following configuration entries in Defaults Configuration (G8WB.CFG) may be used to control batch execution:

```
[G8WB-TRACE]           ; Trace configuration
BATCH-EXEC-CALL      : OFF       ; Trace Interface/procedure calls
BATCH-GLOBAL-LOG    : ON        ; OFF(default)/ON
```

BATCH-GLOBAL-LOG:ON cause the log file to be appended to a consolidated log file \$G8RBLDIRr\BATCH.yyddd.LOG file, where yyddd = julian date.

BATCH-EXEC-CALL:ON cause debug/trace information to be written to the log file.

## 2.2 ENVIRONMENT SETTING

The RHE8 run-time environment will retrieve basic environment settings from the registry. These settings may be overridden at run-time and additional settings may be added through the RHE8ENV.CFG file.

Application programs - modules, TPR's, programs, database procedures etc., are loaded dynamically and the COBDIR environment setting may specify one or more paths where application programs should be loaded from. The search for object files (GNT's) are done left-to-right in the specified paths.

RHE8 will locate the RHE8ENV.CFG file as follows:

- \$RHE8ENV\RHE8ENV.CFG (if RHE8ENV environment setting is present)
- <current directory>\RHE8ENV.CFG
- \$G8RBLDIR\r\RHE8ENV.CFG
- \$G8RBGDIR\RHE8ENV.CFG
- no RHE8ENV.CFG used

If an RHE8ENV.CFG file is found, it's environment settings are loaded and RHE8 will look no further. The RHE8ENV.CFG is a simple text file with one or more environment settings, ex.

```
AP1=F:\APP1
AP2=F:\APP2
SUB=F:\SUB
DBPROC=F:\DBPROC
COBDIR=%SUB%;%DBPROC%;%COBDIR%
COBDIR=%AP1%;%AP2%;%COBDIR%
G8RBLDIR=C:\RHE8\L
```

Existing environment variables must be expanded using %symbol% technique.

## 2.3 FILE MAPPING

Conventional files - GFRC, UFAS etc., are accessed through the GCOS8 file-code. The file-code is treated as a 'logical name'.

### Default

Files are by default accessed through their 'logical name' under the current directory.

Ex. GCOS8

```
$ prmf I1,R,S,UMC/myfile
$ sysout P*
```

Ex. RHE8

```
read file I1 under current directory
write file SYSOUT under current directory
```

### Mapping

Mapping to a specific filepath may be done through environment settings using the 'logical name', ex.

```
SET I1=f:\data\i1.dat
SET SYSOUT=f:\data\p1.sysout.txt
```

The setting may include symbols - either %symbol% or \$symbol, however if symbols are defined within RHE8ENV.CFG you must use \$symbol technique (no defined at time of setting), ex.

```
SET I1=%mydata%\i1.dat
SET I1=$app1data\i1.dat
```

\$symbol can only be used as prefix, and will be resolved at run-time.

## Special file-codes

The I\* file-code use SYSIN as logical name.

The P\* file-code use SYSOUT as logical name.

The \$\$ file-code is not used, instead the execute report is written to a log file.

Any file-code which is a COBOL reserved word will have an X appended in the logical name, ex. IN becomes INX, OF becomes OFX.

PSM files are mapped slightly different, ex.

```
PSM-file is named TEST.PSM

TESTSA is logical name for input file
TESTSZ is logical name for output file
```

## Temporary files

RHE8 treat all files as permanent files. The \$ FILE feature of GCOS8 is not supported, however when building job streams you can simply include a file delete operation.

## Lud

The Logical Unit Designator (lud) feature of GCOS8 is not supported, however you can obtain similar behavior through environment settings, ex.

```
SET LUDX1=$app1data\myfile.dat
SET O1=%ludx1%
RUN PRG p1
SET I1=%ludx1%
RUN PRG p2
```

## Sort files

Sort work files - either PSM or embedded COBOL sort, are automatically allocated (create/delete) in the path specified in TEMP environment setting.

## 2.4 RUNNING UTILITIES

You use the RUN command to launch RHE8 utilities. The command line syntax is as follows:

```
RUN command argument(s) [=output]
```

The optional =output argument may be used to redirect the log file to the specified filepath. By default the log file is written to \$G8RBLDIR\batch.log.

### Print Expansion

```
RUN PRCEXP print-file[.ext] [report-file]
```

Expand Report-code or Report-writer print-file into separate report files. Default extension for print-file is DAT. Expanded report-file(s) use extension Dnn.DAT, where nn = report-code. Default filename is input filename.

```
RUN PRCEXPI
```

Execute Print expansion utility in interactive mode.

### PSM sort

```
RUN PSM psm-file [_List-source]
```

Execute a PSM sort.

```
RUN PSMI
```

Execute PSM sort utility in interactive mode.

### PSW handling

```
RUN PSWS nn+|- [...]
```

Set or reset one or more PSW switches, where nn = 00-35 or 99 (all), and + = set switch ON and - = set switch OFF.

```
RUN PSWT nn
```

Test status of PSW switch, where nn = 00-35. If ON, ErrorLevel is set to 1 (for use in job streams).

```
RUN PSWI
```

Execute PSW utility in interactive mode.

## Interactive IDS-II

RUN IIDS subschema FILE filename

Execute interactive IDS-II utility with DML script file.

RUN IIDSI

Execute interactive IDS-II utility in interactive mode.

## Convert file

RUN G8WFL profile[.g8p] [inputfile[.g8f] outputfile[.dat]]

Convert a GCOS8 file to a native file. The profile must be created in interactive mode. The profile may specify input and output file details, however these may be overridden through command line arguments.

RUN G8WFLI

Execute PSW utility in interactive mode.

## Load database

RUN DBLOAD schemaname load-file[.dat] [\_Update [Only]] [\_Force]  
[\_Trace] [\_1|\_2|\_3]

Load a database - whole or parts of, from a GCOS8 unload file. The default load mode is INITIALIZE, meaning wipe existing database and load a new one using INSERT.

\_Update may be used to request a partial load, meaning existing database is preserved and records added either with INSERT or UPDATE.

\_Force may be used to override schema versions checks (should be used with care).

\_Trace may be used to generate a trace of the load operation. This feature may be used to debug errors occurring during load, however keep in mind the log file may become very large.

\_1|\_2|\_3 may be used to request a partial (phase) load of a complete database load. Phase 1 is load of all records. Phase 2 is load (update) of all sets. Phase 3 is load (update) of phantom pointers. Must be used in correct sequence.

RUN DBLOADI

Execute Load database utility in interactive mode.

## **Log message**

RUN LOG some message

Write a message in the consolidated log file.

## **Batch**

RUN BATCH command-file[.cvt]

Execute a RHE8 command file. Default extension is CBT.

An RHE8 command file is similar to a Windows command file, except only RHE8 commands and SET commands may be used. Commands within a RHE8 command file are not prefixed with RUN.

## **Configuration**

RUN CONFIG

Execute the RHE8 configuration utility.

## **Logenv**

RUN LOGENV

Write the environment configuration to RHE8ENV.LOG in current directory.

## 2.5 JCL/JOB STREAMS

The GCOS Rehosting Express run-time environment does not support GCOS8 JCL, however existing GCOS8 JCL may easily be migrated into command file scripts. You may also consider using scheduling facilities offered by or for the Windows run-time environment.

### Example JCL migration

GCOS8 JCL example for print program accessing an IDS-II and producing a report using Report-write, which is then printed via sysout.

```
$ IDENT ...
$ SELECT umc/obj/p1
$ SELECT umc/subs/subs1/cstar
$ EXECUTE
$ DATA I*
$ SELECT umc/data/par1
$ FILE LS,X1S
$ PRMFL 1*,R,R,umc/sch/sch1/1star
$ PRMFL F1,R,R,umc/db/sch1/f1area
$ PRMFL F2,R,R,umc/db/sch1/f2area
$ CONVER
$ FILE IN,X1R
$ SYSOT OT,ORG
$ENDJOB
```

Equivalent command file for RHE8 run-time environment.

```
rem My example batch stream
SET SYSIN=$app1\files\par1.txt
SET LS=$app1\files\output.dat
rem Execute P1
RUN PRG p1
rem Expand report-file and print it
RUN PRCEXP $app1\files\output wrk
PRINT wrk.D10
DELE wrk.D10
```

## Example conditional

An simplified example of a migrated GCOS8 JCL using conditional JCL and Program-Switch-Word.

```
rem My example batch stream
rem Reset all switches
RUN PSWS -99
rem Execute P1
RUN PRG p1
rem Test switch 21
RUN PSWT 21
if errorlevel 1 goto label2
:label1
rem Execute P2
RUN PRG p2
:label2
```



### **3. RDBMS ENVIRONMENT**

GCOS8 Rehosting Express use a standard RDBMS to host the IDS-II database emulation. Following RDBMS software is supported:

- INFORMIX IDS.2000 Version 9.2
- ORACLE 8/8i/9i
- IBM DB2 UDB Version 7/8/8.1
- Microsoft SQL Server 7.0/2000

The IDS-II schema(s) and subschema(s) are translated and validated in the development environment and hereafter copied into the RHE8 run-time environment. All IDS-II schemas are hosted in the same database, which is created using the SQL definition files generated during schema validation. Existing IDS-II databases on GCOS8 are unloaded using the generated unload, and subsequently loaded in the RHE8 run-time environment using the generated load programs. The IDS-II Configuration must be amended before you can load and utilize the database(s).

GCOS8 Rehosting Express use IBM DB2 RDBMS to host Interel/RFM databases. Interel/RFM is not fully compatible with IBM DB2, so you need first to migrate the SQL definition files in the development environment. Once the IBM DB2 databases are created, you may generate unload scripts for INFOEDGE and unload existing Interel/RFM databases on GCOS8. Next you may use the import utility to load the database in the RHE8 run-time environment. The INTEREL/RFM Configuration must be amended before you can load and utilize the database(s).

### **IDS-II Techniques**

The following implementation techniques are used for rehosting of IDS-II databases on any of the supported RDBMS.

#### **Schema components**

Schema records are implemented as an SQL table with equivalent data fields. Each record is uniquely identified by a database key, which is added as the first field in a table. The database key is the primary key for a table. A range of database keys are reserved for each record type, and the range is used to identify record type.

Schema keys - calc, index and alternate, are implemented as SQL indexes on the table.

Schema sets are implemented as an additional data field in the table (members only). The set relationship is the database key of the set owner, plus a sequence number for non-sorted sets. An SQL index is build for each set membership. Foreign key constrains are used to maintain integrity for the set, ex. a member cannot exist without it's owner.

Schema areas are implemented as an additional data field in the table (logical area association).

The same component names are used where possible, otherwise names are generated. RDBMS may impose limitation on name length.

All IDS-II databases are hosted in the same RDBMS database instance. It is not possible to use multiple database instances within the same RHE8 run-time environment.

## Database key allocation

Database key (dbkey) allocation for GCOS8 is defined for an area and the dbkey allocation and optional reserve reflect the number of records that can be stored in an area. As records are deleted, dbkeys are reused.

Dbkey allocation for RHE8 is defined for a record and the dbkey allocation reflect the number of records (of this type) that can be stored during the life-cycle of the database. As records are deleted, dbkeys are NOT reused. To prevent running out of dbkeys, you must consider the dbkey allocation carefully before loading your production database for use in RHE8 run-time environment.

The schema configuration control the dbkey allocation for the schema. The default is 2100000, meaning each record within the schema is thus assigned a dbkey range of 2.1 million. You may configure any other number, or CONFIG to control allocation more explicitly. A separate tag is used to specify DBkey allocation details, this tag must be name <schemaname>-INTERVAL, ex.

```
[DIMS-INTERVAL]
DEFAULT      : 2100000      ; Default allocation
COURSE      : 40000000     ; Specific allocation
SCEHDULE    : 80000000     ; Specific allocation
```

Whenever a schema (a table) is dropped and re-created, the dbkey range is reset.

You may use DBUNLOAD and DBLOAD to re-gain lost dbkeys. It is possible re-gain lost dbkeys using an own developed SQL script, however you must keep in mind set relations, which are using the owner dbkey.

## Hidden fields

CHAR fields in an IDS-II schema can contain 'hidden' fields, and such fields MUST always be converted during unload/load if the are in packed or binary formats, otherwise data will be incorrect and for some RDBMS software considered illegal.

NULL characters (low-values) are not allowed in CHAR fields for some RDBMS software. Where applicable you can use the NULL-FIX settings to force null characters to spaces (works both for load and application programs). Some RDBMS software will not allow packed or binary data in CHAR, and you then need to change the format to DISPLAY. This may affect the size of the CHAR field in schema/subschema.

The PDATA-TRACE setting may be used to document presence of packed or binary data in CHAR fields (works both for load and application programs).

## Phantom pointers

RHE8 will automatically adapt dbkeys for records and sets, however it cannot adapt phantom pointers (data field with dbkey pointer) automatically. If phantom pointers are used, you need to amend the schema configuration rules before you generate unload/load programs.

## Area use

For GCOS8 you could store records in multiple areas through area selection, using a DB parameter of DB procedure.

For RHE8 areas are considered logical and an attribute of a table, however functionally area selection is performed equivalent to GCOS8. You may however suppress all area selection via the AREA-SELECT setting. If suppressed, all records are stored in the first realm (from WITHIN clause), no area attribute is associated with the table and DB parameters or procedures are ignored.

Note: Access to other areas (non-first) are still possible, however these are all empty.

## Space allocation

For GCOS8 the physical space allocation was controlled via schema DMCL and the database could (or had to) be spread amongst multiple physical files.

For RHE8 all IDS-II databases (all schemas) are rehosted in one instance of the target RDBMS software, and the physical space allocation can be left entirely to the target RDBMS software or partially controlled by you.

The Location database setting for a schema controls space allocation as follows:

CURRENT = use default tablespace for all tables  
SCHEMA = create and use a schema specific tablespace  
AREA = create and use an area specific tablespace

A record can ONLY be placed in one tablespace, and the first area, i.e. WITHIN clause, is always used. Area-selection is not affected by this.

Note: For a production environment you should consider use of SCHEMA or AREA location, or a combination of these.

## 3.1 IDS-II CONFIGURATION

The IDS-II Configuration file (\$G8RBDDIR\DBMS.CFG) is used to define the schemas and subschemas available to your applications and how RHE8 interface with the hosting RDBMS. The RHE8 run-time environment contains a copy of the run-time support files and programs, which have been created in your development environment.

## Schema configuration

You should simply copy/paste the existing schema configuration entries from your development environment. This will ensure the schema numbers are the same in both environments. Following illustrate a simple schema configuration:

```
[SCHEMA-SECTION]
;
;          Number Location Location Dbkey
;          (1-99) Schema  database allocation per table
;          CURRENT  DEFAULT (2100000
dbkeys)
;
;          SCHEMA  nnnnnnnnn
;          AREA    CONFIG (separate tag)
DIMS      : 23    CURRENT CURRENT  DEFAULT ; DIMS8
TELEX8    : 24    CURRENT CURRENT  DEFAULT ; DIMS8
```

You then use the 'Move schema' function in the development environment to copy all run-time support files into the RHE8 run-time environment (\$G8RBDDIR).

## Subschema configuration

You should simply copy/paste the existing subschema configuration entries from your development environment. This will ensure the subschema short names are the same in both environments. Following illustrate a simple schema configuration:

```
;          Short  Location  Auto
;          name   Subschema generated
[SUBSCHEMA-SECTION]
CLASS      : CLASS   CURRENT ; example
SS-CLASS   : 26SCHS  CURRENT -AUTO ; example
```

You then use the 'Move subschema' function in the development environment to copy all run-time support files into the RHE8 run-time environment (\$G8RBDDIR).

## 3.2 INFORMIX RDBMS

The following relate to INFORMIX IDS.2000 Version 9.2 as target RDBMS only.

### Techniques

All records have a unique dbkey field, which is used as primary key. The dbkey field is implemented as a SERIAL datatype with a min/max range as configured for the schema.

During load of database the GCOS8 dbkey value is remapped to the new dbkey value assigned during load. This remapping apply to set and phantom pointers. When accessing records through Calc or Index keys or sets, RHE8 will use a hint for the appropriate key, and thus force Informix optimizer to use the correct index for access. Sets are implemented as foreign keys in the member record(s), and the dbkey of the owner is use to enforce a constraint for the set relationship. For manually sorted sets - insertion first/last/next/prior, an additional order field is maintained.

### Configuration

The SQL-SECTION of the IDS-II Configuration file (DBMS.CFG) defines Informix as the host for IDS-II databases and various settings for interfacing with Informix. You should simply copy/paste the existing SQL-SECTION entries from your development environment and change interface settings as required.

```
[SQL-SECTION]
; Enable/Disable RHE8
SQL-IMPLEMENTATION : INFORMIX ; INFORMIX
; ; OFF (ISAM)
SERVER : ol_instance ; Name of server
DATABASE : inf_ids2 ; Name of database
USERID : informix ; Name of user
PASSWORD : informix ; Password of user
NULL-FIX : ON ; Fix null in char
OFF(default)/ON
NULL-TRACE : OFF ; Trace null fix OFF(default)/ON
DEBUG : OFF ; SQL Debug-log OFF(default)/ON
EXPLAIN : OFF ; SQL explain OFF(default)/ON
WAIT-LOCK : ON ; SQL wait lock ON(default)/OFF
ISOLATION : HIGH ; Isolation
HIGH(default)/LOW/OFF
.LOCK-SUPPORT : ON ; .LOCK ON(default)/OFF
AREA-SELECT : ON ; ON(default)/OFF
DECIMAL-POINT-COMMA : OFF ; OFF(default)/ON
MAX-LOAD-UNIT : 10000 ; commit every n (default 500)
ESQL-C : ON ; ON(required)
ORDER-PROGRAM : ON ; Enforce non-sorted set
ordering On(default)/OFF
CASCADE-DELETE : ON ; Use constaint rules
ON(Default)/OFF
CHECKSUM-CHECK : OFF ; Debug OFF(default)/ON/ALWAYS
```

## Interface settings

SERVER setting designate the name of the Informix IDS server instance. Default naming is ol\_XXXXX. The name cannot exceed 9 characters.

DATABASE setting designate the name of the Informix IDS database. It is recommended to use the default name = inf\_ids2.

USERID and PASSWORD setting designate which identification to use when establishing sessions with Informix. Use the default Informix userid with appropriate password.

## Optional settings

NULL-FIX:ON setting may be used force NULL characters in CHAR type fields to SPACE. NULL characters, i.e. low-values, are not allowed in CHAR type fields.

WAIT-LOCK:ON is default, i.e. application will not abort until a lock-wait time-out occur. The time-out period is controlled via settings in DEFAULT-SECTION as follows:

BATCH-WAIT-TIME : 180 ; Wait time in seconds, default 180  
TP-WAIT-TIME : 30 ; Wait time in seconds, default 30

ISOLATION:HIGH is default. For batch: If HIGH, it is changed to LOW, except with QUERY access mode, then it is changed to OFF. The equivalence for Informix is as follows:

HIGH = REPEATABLE READ  
LOW = COMMITTED READ  
OFF = DIRTY READ

.LOCK-SUPPORT:ON is default, and a .LOCK service call will cause the current record to be 'locked for update'. Alternatively .LOCK service calls are ignored (OFF).

DECIMAL-POINT-COMMA: OFF is default, ON must be specified if Informix is configured to treat comma as decimal point.

MAX-LOAD-UNIT: 500 is default, may be used to control how often commits are taken during load of database.

## Debug settings

NULL-TRACE:ON may be used to log information on NULL-FIX activity on the execution log (batch or TP8). Statistics show the field-name and a corresponding count for null-fix operations.

DEBUG:ON may be used to log a substantial amount of information about ESQL activity on the %g8rbldir\DBESQL.LOG file. This feature must be used with caution since it degrades performance and produces a very large log file.

EXPLAIN:ON may be used to set explain mode for all SELECT operations performed by RHE8. The explain information is logged within RDBMS environment, ref. Informix documentation for details.

### **Development settings**

These settings must be the same as for the development environment, because these affect the generated SQL definitions and unload/load programs.

AREA-SELECT: ON is default, OFF may be used to ignore area selection. When OFF, any area selection procedures are ignored and records are always stored/retrieved from the first realm. All other realms are treated as empty.

ORDER-PROGRAM: ON is default and used to ensure non-sorted sets are maintained in the appropriate FIRST/LAST/NEXT/PRIOR order. OFF may be used to enable a simpler and more efficient emulation (dbkey order), which however is not compatible with GCOS.

CASCADE-DELETE: ON is default and will use SQL constraint rules to enforce owner/membership relations, which is effective both for IDS-II and direct SQL database access. OFF may be used to suppress use of constraint rules and emulation of ERASE ALL MEMBERS.

### **Database setup & load**

The SQL definition files are located under %g8rbddir\schema as follows:

- nnMTABL.SQL - create all tables and primary index
- nnMINDX.SQL - create all indexes (Calc, Index key)
- nnMCONS.SQL - create set indexes and constraints

- nnDTABL.SQL - create all tables and primary index
- nnDINDX.SQL - create all indexes (Calc, Index key)
- nnDCONS.SQL - create set indexes and constraints

where nn = schema number

Creation of an empty database can be done with the DBACCESS utility, when executed from the ol instance command prompt, ex.

```
dbaccess inf_ids2 <g8rbddir>\schema\23MTABL.SQL
dbaccess inf_ids2 <g8rbddir>\schema\23MINDX.SQL
dbaccess inf_ids2 <g8rbddir>\schema\23MCONS.SQL
```

When Database location = SCHEMA, two additional command files (BAT) are available:

nnCSPACE.BAT - Create SBSPACE for schema  
nnDSPACE.BAT - Drop SBSPACE for schema

When Database location = AREA, two additional command files (BAT) are available:

nnCSPACE.BAT - Create SBSPACE for all areas within schema  
nnDSPACE.BAT - Drop SBSPACE for all areas within schema

Note: You need to change <path> information and space allocation for each DBSPACE, before you execute the command file.

The unload and load programs must be generated in the development environment, before you copy the schema to the RHE8 run-time environment.

Transfer the nnHDBU.CBL to GCOS8, and optionally the subschema (may already be present on the host). Next compile and execute the nnHDBU program to unload the whole or parts of the host database. Next transfer the unload file(s) to RHE8 run-time environment (UFT/FTP ascii file).

Before loading the database, you need to prepare the Informix database as follows:

1. Modify the nnCSPACE.BAT and execute this to create the DBSPACE(s) for the schema. This does not apply for Database location = CURRENT.
2. Set database in 'no-logging' mode using ONDBLOG utility, ex. ONDBLOG nolog inf\_ids2
3. Create tables for the schema using DBACCESS utility, ex. DBACCESS inf\_ids2 <g8rbddir>\schema\23MTABL.SQL
4. Load the database using DBLOAD utility, ex. DBLOAD dims dumpdims. \_nolog

Repeat this step for each unload file, if multiple files were used for GCOS8 unload.

5. Create indexes and constrains for the schema using DBACCESS utility, ex.

DBACCESS inf\_ids2 \<g8rbddir>\schema\23MINDX.SQL  
DBACCESS inf\_ids2 \<g8rbddir>\schema\23MCONS.SQL

6. Set database in 'logging' mode using ONDBLOG utility, ex. ONDBLOG unbuf inf\_ids2

The request does not become effective until a level-0 backup is performed for the database.

7. Do a complete (level-0) backup of the database using the ONBAR utility, ex. ONBAR -b -L 0

The DBLOAD utility will execute in two phases if the unload file contains both record and set information (default). In Phase 1, only records are processed and each record is inserted in the database. A serial number (new dbkey) is assigned to each record, and GCOS8 db-key is mapped to the serial number (new dbkey) using a table

snn\_dbk\_map. In Phase 2, only sets are processed and each set membership causes the corresponding record to be updated with the serial number (new dbkey) of the owner.

If you need to re-run the DBLOAD utility, it is recommended to drop all tables or the tables involved, and re-create these, before running DBLOAD. This way serial number ranges are reset.

The DBLOAD utility may be directed to perform only a single phase, ex.

```
DBLOAD dims dumpdims. _1
DBLOAD dims dumpdims. _2 _u
```

## Logging

For on-line TP8 operation, the database must be protected with logging enabled. By default, logging files are backed up automatically (ALARMPROGRAM setting). Should a disaster occur, the database can be fully recovered to a consistently state near the time of disaster.

For batch operation, the database may or may not be protected with logging enabled, depending on operation mode and the application.

- Batch concurrent with on-line: Logging **MUST** be enabled, and the application must either take frequent checkpoints, or use the 'auto' database mode.

Warning: Should the batch program apply incorrect changes to the database, you are **NOT** able to restore the database without potential loss of changes made through on-line applications.

- Batch stand-alone: Logging may be enabled, however, the application must either take frequent checkpoints, or use the 'auto' database mode, otherwise the LOCKS resource may be exhausted.
- Batch stand-alone/heavy: Logging may be disabled, however, be aware the database **MUST** be backed up (level-0) whenever the logging mode is changed. The application **MUST** use the 'nolog' database mode, otherwise it will be aborted.

For batch, use 'exclusive' database mode, if you want to prevent others (batch or TP8) from accessing the database.

For batch, use 'query' database mode, if the application is only reading from the database. When running concurrent with TP8, the batch application will neither reserve locks or respect locks on records. This mode is quite similar to QUERY allocation mode on GCOS8.

## Backup

Informix IDS Server provides following levels of backup:

- Level-0 - A full backup of the database

- Level-1 - An incremental backup of changed pages (since level-0)
- Level-2 - An incremental backup of changed pages (since level-1)
- Logical Logs - A backup of all changes since last backup

A level-0 backup should be performed

- after initial load of database
- after changing logging mode
- at frequent intervals, i.e. once a week, to limit the size of level-1/2 backup and logical logs

A level-1/2 backup should be performed

- after completion of on-line operation day
- after completion of large batch operations

Note: If level-2 is requested and no level-1 exist since last level-0 backup, the backup is treated as level-1.

Backup operations are requested with the ONBAR utility, ex.

```
ONBAR -b -L 0
```

```
ONBAR -b -L 2
```

Note: Logical logs are backed up automatically, either when filled or when a level-n backup is requested.

## Restore

Informix IDS Server perform automatic recovery should the instance or the server machine fail. For hardware failures, please refer to Informix documentation on how to restore and recover the database to a consistent clean-point, just before the failure occurred.

The need for an operational restore may arise, if an application has logical errors and incorrect updating of the database has occurred, or you need to re-run a batch application, which have partially committed database updates (checkpoint or auto) or was run with logging disabled.

By default, a restore request will perform restore and recovery to the most recent consistent clean-point. To restore to a specific operational point, i.e. just before running a batch application, a restore request must be specified with a data/time stamp equivalent to the specific operational point.

Note: To ensure proper documentation of operational point and associated date/time stamp, we recommend using BATLOG command to log a meaningful message in the consolidated batch.log (ref. Batch above). Ex.

```
BATLOG Ready to run XXXX program - backup clean-point
```

Restore operations are requested the ONBAR utility, ex.

```
ONBAR -r
```

```
ONBAR -r -t "yyyy-mm-dd hh:mm:ss"
```

## **Backup - Wrapper**

A Backup Wrapper utility (DBBCK) may be used to launch database backups and record information about associated time stamps.

```
DBBCK backup-set-name [0|1|2|W]
```

backup-set-name = a meaningful text to identify the backup

0 = Full backup (level-0)

1 = incremental backup (changes since level-0)

2 = incremental backup (changes since level-1)

W = Complete backup (level-0)

Default is 2

The backup information is recorded in %8rbddir\BACKUP.CFG.

## **Restore - Wrapper**

A Restore Wrapper utility (DBRES) may be used to launch database restores without knowledge of actual time stamps.

```
DBRES backup-set-name|L|Last-good-backup
```

backup-set-name = the identification used during backup

L = the last-good-backup complete

The associated time stamp is used to launch a restore to the point right after the backup was completed.

Note: All restores are cold (Informix shutdown and restarted) and the database is left in Quecient mode after the restore completes.

## 3.3 ORACLE RDBMS

The following relate to ORACLE 8/8i/9i as target RDBMS only.

The Oracle Database may be installed either on the same Windows NT/2000/2003 server as RHE8, or on any other server, including Linux and Unix platforms. When installed on another server, you must install the Oracle Client on the RHE8 server and setup appropriate connect string(s) for RHE8.

### Techniques

All records have a unique dbkey field, which is used as primary key. The dbkey field is implemented as a SERIAL datatype with a min/max range as configured for the schema.

During load of database the GCOS8 dbkey value is remapped to the new dbkey value assigned during load. This remapping apply to set and phantom pointers. When accessing records through Calc or Index keys or sets, RHE8 will use a hint for the appropriate key, and thus force Oracle optimizer to use the correct index for access. Sets are implemented as foreign keys in the member record(s), and the dbkey of the owner is use to enforce a constraint for the set relationship. For manually sorted sets - insertion first/last/next/prior, an additional order field is maintained.

RHE8 always use OPTIMIZER\_MODE = FIRST\_ROWS.

### Configuration

The SQL-SECTION of the IDS-II Configuration file (DBMS.CFG) defines Informix as the host for IDS-II databases and various settings for interfacing with Oracle. You should simply copy/paste the existing SQL-SECTION entries from your development environment and change interface settings as required.

```
[SQL-SECTION]
;                               Enable/Disable RHE8
SQL-IMPLEMENTATION : ORACLE      ; Oracle 8/8i/9i
;                               ; OFF (ISAM)
DATABASE           : ora_ids2    ; Database name (TNS name)
USERID             : scott       ; Name of user
PASSWORD          : tiger       ; Password of user
NULL-FIX          : ON          ; Fix null in char
OFF(default)/ON
NULL-TRACE        : OFF         ; Trace null fix OFF(default)/ON
DEBUG             : OFF         ; SQL Debug-log OFF(default)/ON
ISOLATION         : LOW        ; Isolation LOW(default)/HIGH
.LOCK-SUPPORT     : ON         ; .LOCK ON(default)/OFF
AREA-SELECT       : ON         ; ON(default)/OFF
DECIMAL-POINT-COMMA : OFF      ; OFF(default)/ON
MAX-LOAD-UNIT     : 10000      ; commit every n (default 500)
EXPLAIN           : OFF        ; OFF(required)
WAIT-LOCK         : ON         ; ON(required)
ESQL-C            : OFF        ; OFF(required)
ORDER-PROGRAM     : ON         ; Enforce non-sorted set
```

```
ordering On(default)/OFF
CASCADE-DELETE      : ON           ; Use constaint rules
ON(Default)/OFF
CHECKSUM-CHECK      : OFF           ; Debug OFF(default)/ON/ALWAYS
```

## Interface settings

DATABASE setting designate the name of the Oracle database (TNS name).

USERID and PASSWORD setting designate which indentification to use when establishing sessions with Oracle. Use the default Scott userid with appropriate password, or any other userid.

## Optional settings

NULL-FIX:ON setting may be used force NULL characters in CHAR type fields to SPACE. NULL characters, i.e. low-values, are not allowed in CHAR type fields.

WAIT-LOCK:ON is mandatory, i.e. application will not abort until a lock-wait time-out occur. The time-out period is controlled via settings in DEFAULT-SECTION as follows:

```
BATCH-WAIT-TIME : 180 ; Wait time in seconds, default 180
TP-WAIT-TIME    : 30  ; Wait time in seconds, default 30
```

ISOLATION:LOW is default, however this may be changed to HIGH as required. The equivalence for Oracle is as follows:

```
HIGH = SERIALIZABLE
LOW  = COMMITTED READ
```

Note: Oracle does not support query/nolog mode of operation and will use the isolation level low/high as configured.

.LOCK-SUPPORT:ON is default, and a .LOCK service call will cause the current record to be 'locked for update'. Alternatively .LOCK service calls are ignored (OFF).

DECIMAL-POINT-COMMA: OFF is default, ON must be specified if Oracle is configured to treat comma as decimal point.

MAX-LOAD-UNIT: 500 is default, may be used to control how often commits are taken during load of database.

## Debug settings

NULL-TRACE:ON may be used to log information on NULL-FIX activity on the execution log (batch or TP8). Statistics show the field-name and a corresponding count for null-fix operations.

DEBUG:ON may be used to log a substantial amount of information about ESQLE activity on the %g8rbldir\DBESQL.LOG file. This feature must be used with caution since it degrades performance and produces a very large log file.

## Development settings

These settings must be the same as for the development environment, because these affect the generated SQL definitions and unload/load programs.

AREA-SELECT: ON is default, OFF may be used to ignore area selection. When OFF, any area selection procedures are ignored and records are always stored/retrieved from the first realm. All other realms are treated as empty.

ORDER-PROGRAM: ON is default and used to ensure non-sorted sets are maintained in the appropriate FIRST/LAST/NEXT/PRIOR order. OFF may be used to enable a simpler and more efficient emulation (dbkey order), which however is not compatible with GCOS.

CASCADE-DELETE: ON is default and will use SQL constraint rules to enforce owner/membership relations, which is effective both for IDS-II and direct SQL database access. OFF may be used to suppress use of constraint rules and emulation of ERASE ALL MEMBERS.

## Database setup & load

The SQL definition files are located under %g8rbddir\schema as follows:

- nnMTABL.SQL - create all tables and primary index
- nnMINDX.SQL - create all indexes (Calc, Index key)
- nnMCONS.SQL - create set indexes and constraints

- nnDTABL.SQL - create all tables and primary index
- nnDINDX.SQL - create all indexes (Calc, Index key)
- nnDCONS.SQL - create set indexes and constraints

where nn = schema number

Creation of an empty database can be done with the SQL\*Plus utility, ex.

(first connect to appropriate database)

```
start <g8rbddir>\schema\23MTABL.SQL
start <g8rbddir>\schema\23MINDX.SQL
start <g8rbddir>\schema\23MCONS.SQL
```

When Database location = SCHEMA, two additional SQL definition files are available:

- nnCSPACE.SQL - Create TABLESPACE for schema
- nnDSPACE.SQL - Drop TABLESPACE for schema

When Database location = AREA, two additional SQL definition files are available:

nnCSPACE.SQL - Create TABLESPACE for all areas within schema  
nnDSPACE.SQL - Drop TABLESPACE for all areas within schema

Note: You need to change datafile information and space allocation for each TABLESPACE, before you execute the SQL definition file.

The unload and load programs must be generated in the development environment, before you copy the schema to the RHE8 run-time environment.

Transfer the nnHDBU.CBL to GCOS8, and optionally the subschema (may already be present on the host). Next compile and execute the nnHDBU program to unload the whole or parts of the host database. Next transfer the unload file(s) to RHE8 run-time environment (UFT/FTP ascii file).

Before loading the database, you need to prepare the Oracle database as follows:

1. Modify the nnCSPACE.SQL and execute this to create the TABLESPACE(s) for the schema. This does not apply for Database location = CURRENT.
2. Set ARCHIVE LOGGING off using Oracle Console or command.
3. Create tables for the schema using SQL\*Plus utility, ex.

```
<connect to the appropriate database>  
START <g8rbddir>\schema\23MTABL.SQL
```

4. Load the database using DBLOAD utility, ex.

```
DBLOAD dims dumpdims.
```

Repeat this step for each unload file, if multiple files were used for GCOS8 unload, however remember to use \_update for all but the first load file, ex.

```
DBLOAD dims dumpdims1.  
DBLOAD dims dumpdims2. _u  
DBLOAD dims dumpdims3. _u
```

5. Create indexes and constrains for the schema using SQL\*Plus utility, ex.

```
<connect to the appropriate database>  
START <g8rbddir>\schema\23MINDX.SQL  
START <g8rbddir>\schema\23MCONS.SQL
```

6. Set ARCHIVE LOGGING on using Oracle Console or command.
7. Do a complete backup of the database using Oracle Console or command.

The DBLOAD utility will execute in two phases if the unload file contains both record and set information (default). In Phase 1, only records are processed and each record is inserted in the database. A serial number (new dbkey) is assigned to each record, and GCOS8 db-key is mapped to the serial number (new dbkey) using a table snn\_dbk\_map. In Phase 2, only sets are processed and each set membership causes the corresponding record to be updated with the serial number (new dbkey) of the

owner. An optional Phase 3 is triggered if phantom pointers are defined and involved records are detected in Phase 1 (record processing).

If you need to re-run the DBLOAD utility, it is recommended to drop all tables or the tables involved, and re-create these, before running DBLOAD. This way serial number ranges are reset.

The DBLOAD utility may be directed to perform only a single phase, ex.

```
DBLOAD dims dumpdims. _1
DBLOAD dims dumpdims. _2 _u
```

## **Logging**

For on-line TP8 operation, the database must be protected with archive logging enabled. By default, logging files are backed up automatically. Should a disaster occur, the database can be fully recovered to a consistently state near the time of disaster.

For Batch concurrent with on-line: The application must either take frequent checkpoints, or use the 'auto' database mode to prevent TP from waiting for resources held by batch application.

For Batch stand-alone: The application must either take frequent checkpoints, or use the 'auto' database mode, otherwise the LOCKS resource may be exhausted.

## 3.4 IBM DB2 RDBMS

The following relate to IBM DB2 UDB Version 7/8/8.1 as target RDBMS only.

The DB2 Database may be installed either on the same Windows NT/2000/2003 server as RHE8, or on any other server, including Linux and Unix platforms. When installed on another server, you must install the DB2 Client on the RHE8 server and configure the database(s) for DB2.

### Techniques

All records have a unique dbkey field, which is used as primary key. The dbkey field is implemented as a SERIAL datatype with a min/max range as configured for the schema.

During load of database the GCOS8 dbkey value is remapped to the new dbkey value assigned during load. This remapping apply to set and phantom pointers. When accessing records through Calc or Index keys or sets, RHE8 will use a hint for the appropriate key, and thus force DB2 optimizer to use the correct index for access. Sets are implemented as foreign keys in the member record(s), and the dbkey of the owner is use to enforce a constraint for the set relationship. For manually sorted sets - insertion first/last/next/ prior, an additional order field is maintained.

RHE8 always use QUERY OPTIMIZATION = 0.

### Configuration

The SQL-SECTION of the IDS-II Configuration file (DBMS.CFG) defines Informix as the host for IDS-II databases and various settings for interfacing with DB2. You should simply copy/paste the existing SQL-SECTION entries from your development environment and change interface settings as required.

```
[SQL-SECTION]
;
; Enable/Disable RHE8
SQL-IMPLEMENTATION : DB2 ; IBM DB2 UDB 7/8
; ; OFF (ISAM)
DATABASE : db2_ids2 ; Name of database
USERID : db2admin ; Name of user
PASSWORD : db2admin ; Password of user
NULL-FIX : OFF ; Fix null in char
OFF(default)/ON
; Null IS allowed in CHAR fields
NULL-TRACE : OFF ; Trace null fix OFF(default)/ON
PDATA-TRACE : OFF ; Trace 'packed' data in char
OFF(default)/ON
DEBUG : OFF ; SQL Debug-log OFF(default)/ON
EXPLAIN : OFF ; SQL explain OFF(default)/ON
; ON = EXPLAIN MODE YES
; DB2 explain tables MUST be
created first
WAIT-LOCK : ON ; SQL wait lock ON(required)
ISOLATION : HIGH ; HIGH = READ COMMITTED (not
```

```

configurable)
.LOCK-SUPPORT      : ON          ; .LOCK ON(default)/OFF
AREA-SELECT       : ON          ; ON(default)/OFF
DECIMAL-POINT-COMMA : OFF       ; OFF(default)/ON
MAX-LOAD-UNIT     : 10000      ; commit every n (default 500)
ESQL-C            : OFF       ; OFF(required)
ORDER-PROGRAM     : ON          ; Enforce non-sorted set
ordering On(default)/OFF
CASCADE-DELETE    : ON          ; Use constraint rules
ON(Default)/OFF
CHECKSUM-CHECK    : OFF       ; Debug OFF(default)/ON/ALWAYS

```

## Interface settings

DATABASE setting designate the name of the DB2 database.

USERID and PASSWORD setting designate which identification to use when establishing sessions with DB2. Use the default db2admin userid with appropriate password, or any other userid.

## Optional settings

NULL-FIX:ON setting may be used force NULL characters in CHAR type fields to SPACE. NULL characters, i.e. low-values, are allowed in CHAR type fields.

PDATA-TRACE:ON may be used to log information on 'packed' data detections on the execution log (batch or TP8). Statistics show the field-name and a corresponding count for 'packed' data detections.

WAIT-LOCK:ON is required, i.e. application will not abort until a lock-wait time-out occur. The time-out period is controlled via settings in DEFAULT-SECTION as follows:

```

BATCH-WAIT-TIME : 180 ; Wait time in seconds, default 180
TP-WAIT-TIME    : 30  ; Wait time in seconds, default 30

```

ISOLATION:HIGH is required. The equivalence for DB2 is READ COMMITTED. Note: DB2 does not support query/nolog mode of operation and will always use the isolation level high.

.LOCK-SUPPORT:ON is default, and a .LOCK service call will cause the current record to be 'locked for update'. Alternatively .LOCK service calls are ignored (OFF).

DECIMAL-POINT-COMMA: OFF is default, ON must be specified if DB2 is configured to treat comma as decimal point.

MAX-LOAD-UNIT: 500 is default, may be used to control how often commits are taken during load of database.

## Debug settings

NULL-TRACE:ON may be used to log information on NULL-FIX activity on the execution log (batch or TP8). Statistics show the field-name and a corresponding count for null-fix operations.

DEBUG:ON may be used to log a substantial amount of information about ESQL activity on the %g8rbl\_dir\DBESQL.LOG file. This feature must be used with caution since it degrades performance and produces a very large log file.

EXPLAIN:ON may be used to set explain mode for dynamic SQL operations performed by RHE8. The explain information is logged in the DB2 explain tables. Note: The DB2 explain tables MUST be created before you enable this feature. Refer to DB2 documentation for further details.

## Development settings

These settings must be the same as for the development environment, because these affect the generated SQL definitions and unload/load programs.

AREA-SELECT: ON is default, OFF may be used to ignore area selection. When OFF, any area selection procedures are ignored and records are always stored/retrieved from the first realm. All other realms are treated as empty.

ORDER-PROGRAM: ON is default and used to ensure non-sorted sets are maintained in the appropriate FIRST/LAST/NEXT/PRIOR order. OFF may be used to enable a simpler and more efficient emulation (dbkey order), which however is not compatible with GCOS.

CASCADE-DELETE: ON is default and will use SQL constraint rules to enforce owner/membership relations, which is effective both for IDS-II and direct SQL database access. OFF may be used to suppress use of constraint rules and emulation of ERASE ALL MEMBERS.

## Database setup & load

The SQL definition files are located under %g8rbddir\schema as follows:

nnMTABL.SQL - create all tables and primary index  
nnMINDX.SQL - create all indexes (Calc, Index key)  
nnMCONS.SQL - create set indexes and constraints

nnDTABL.SQL - create all tables and primary index  
nnDINDX.SQL - create all indexes (Calc, Index key)  
nnDCONS.SQL - create set indexes and constraints

where nn = schema number

A corresponding set of command files (BAT) are generated, which may be launched from a DB2 command window. Each command file contains a CONNECT, execution of SQL file and a DISCONNECT. Creation of an empty database can be done from a DB2 command window, ex.

```
<g8rbddir>\schema\23MTABL.BAT  
<g8rbddir>\schema\23MINDX.BAT  
<g8rbddir>\schema\23MCONS.BAT
```

When Database location = SCHEMA, two additional SQL definition files are available:

```
nnCSPACE.SQL - Create TABLESPACE for schema  
nnDSPACE.SQL - Drop TABLESPACE for schema
```

When Database location = AREA, two additional SQL definition files are available:

```
nnCSPACE.SQL - Create TABLESPACE for all areas within schema  
nnDSPACE.SQL - Drop TABLESPACE for all areas within schema
```

Note: You need to change datafile information and space allocation for each TABLESPACE, before you execute the command file.

The unload and load programs are generated in the development environment, before you copy the schema to the RHE8 run-time environment.

Transfer the nnHDBU.CBL to GCOS8, and optionally the subschema (may already be present on the host). Next compile and execute the nnHDBU program to unload the whole or parts of the host database. Next transfer the unload file(s) to RHE8 run-time environment (UFT/FTP ascii file).

Before loading the database, you need to prepare the DB2 database as follows:

1. First bind the RHE8 interface to the DB2 database using the command:

```
RUN INITBND
```

2. Modify the nnCSPACE.SQL and execute this to create the TABLESPACE(s) for the schema. This does not apply for Database location = CURRENT.
3. Set 'Retain log files' to NO using DB2 Control Center or DB2 command.
4. Create tables for the schema using DB2 Command window, ex.

```
<g8rbddir>\schema\23MTABL.BAT
```

5. Load the database using DBLOAD utility, ex.

```
DBLOAD dims dumpdims.
```

Repeat this step for each unload file, if multiple files were used for GCOS8 unload, however, remember to use \_update for all but the first load file, ex.

```
DBLOAD dims dumpdims1.
```

```
DBLOAD dims dumpdims2. _u
DBLOAD dims dumpdims3. _u
```

6. Create indexes and constraints for the schema using DB2 Command window, ex.

```
<g8rbddir>\schema\23MINDX.BAT
<g8rbddir>\schema\23MCONS.BAT
```

7. Set 'Retain log files' to RECOVERY or CAPTURE using DB2 Control Center or DB2 command.
8. Do a complete backup of the database using DB2 Control Center or DB2 command.

The DBLOAD utility will execute in two phases if the unload file contains both record and set information (default). In Phase 1, only records are processed and each record is inserted in the database. A serial number (new dbkey) is assigned to each record, and GCOS8 db-key is mapped to the serial number (new dbkey) using a table `snn_dbk_map`. In Phase 2, only sets are processed and each set membership causes the corresponding record to be updated with the serial number (new dbkey) of the owner. An optional Phase 3 is triggered if phantom pointers are defined and involved records are detected in Phase 1 (record processing).

If you need to re-run the DBLOAD utility, it is recommended to drop all tables or the tables involved, and re-create these, before running DBLOAD. This way serial number ranges are reset.

The DBLOAD utility may be directed to perform only a single phase, ex.

```
DBLOAD dims dumpdims. _1
DBLOAD dims dumpdims. _2 _u
DBLOAD dims dumpdims. _3 _u (only phantom pointers)
```

## Logging

For on-line TP8 operation, the database must be protected with archive logging enabled. By default, logging files are backed up automatically. Should a disaster occur, the database can be fully recovered to a consistently state near the time of disaster.

For Batch concurrent with on-line: The application must either take frequent checkpoints, or use the 'auto' database mode to prevent TP from waiting for resources held by batch application.

For Batch stand-alone: The application must either take frequent checkpoints, or use the 'auto' database mode, otherwise the LOCKS resource may be exhausted.

### **3.5 MICROSOFT SQL-SERVER RDBMS**

The following relate to Microsoft SQL Server 7.0/2000 as target RDBMS only.

Under construction ...

## 3.6 INTEREL/RFM

INTEREL/RFM databases are supported through IBM DB2 UDB Database software, both in the development environment and the RHE8 run-time environment.

### Configuration

The Interel Configuration file (\$G8RBDDIR\INTEREL.CFG) is used to define the models/databases and how they map to DB2 databases. You should simply copy/paste the existing schema configuration entries from your development environment. Following illustrate a simple configuration:

```
[MODEL-SECTION]
;Model          Database Userid Password
KONTRATP8Y      : KONTRA8Y DB2ADMIN secret
KONTRATP8T      : KONTRA8Y DB2ADMIN secret
UTBREDU         : UTBREDU  DB2ADMIN secret
```

The original GCOS8 model name must be mapped to a DB2 database (restricted to eight characters) and must also specify the userid and password to be used for logon to the database.

The default section is used to define run-time behavior as follows:

```
[DEFAULT-SECTION]
TP8              : ON          ; TP8 with INTEREL ON/OFF
BATCH            : OFF         ; Batch INTEREL and IDS-II ON/OFF
DEBUG            : ON          ; SQL Exception and connect trace
```

- TP8, ON must be specified if Interel/RFM programs are being called during TP8 operation.
- BATCH, ON should be used if Interel/RFM programs are being called during batch operation. Only required if the main program is NOT using Interel/RFM.
- DEBUG, ON may be used to obtain a trace of exception information on the TP journal or batch log. Exception trace is triggered when SQLCODE is non zero.

### Setup and Load

All database definition, backup and restore activities are done through appropriate DB2 tools (Control Center, Command Center etc.). RHE8 provides a few utilities to aid setup and load of existing Interel/RFM databases on GCOS8.

#### SQL Files

Interel/RFM SQL files are NOT compatible with DB2 and needs to be migrated before they can be used within DB2. Automated SQL file migration can be done with the RHE8 Migrate Command File utility as follows:

```
RUN DB2MCF interel-file[.ext] [db2-file[.ext]]
```

Default extension for interel-file is OQL. Default extension for db2-file is SQL. Default db2-file name is interel-file name.

```
RUN DB2MCFI
```

Execute Migrate utility in interactive mode.

### Setup unload scripts

After you have created the DB2 database, you may use the RHE8 Setup utility to build an Infoedge script for a snapshot unload of an existing Interel/RFM database on GCOS8. The command syntax is as follows:

```
RUN DB2IED model S[etup] _P[ath] path
```

The model identifies your database. The optional `_Path` argument may be used to direct generated script files to another directory. Default is current directory. The script is named `<model>.IQL`.

```
RUN DB2IEDI
```

Execute Setup utility in interactive mode.

### Load database

The unload script will unload each Interel table into a separate file. The sequence number of the table is used for naming of the unload file. Once unloaded, the unload files must be transferred to the RHE8 run-time environment. You keep the same file naming as designated by the unload script.

You may now use the RHE8 Load utility to load one, more or all tables of your database. The load process will first convert the unload file and then launch DB2 to load the table. The command syntax is as follows:

```
RUN DB2IED model I[mport] _T[able] tablename [_I[nsert]|U[pdate]]
                                     _P[ath] path
```

- Model identifies your database.
- `_Table tablename` is optional, and may be used to limit load of only the specified table.
- `_Insert` is optional, and may be used to limit load to new rows. Default is to replace existing table.
- `_Update` is optional, and may be used to load new or update existing rows.
- `_Path path` is optional, should be used if unload files are not in current directory.

## Application setup

DB2 use BIND files for static SQL, and bind files are registered in the DB2 database during compilation. The bind files are placed in LOADLIB\<<model>\<program>.bnd (MFE8) or DEBUG\model\<program>.bnd (NE8) files, together with the program object files. When copying program object files from the development environment to the RHE8 run-time environment, you must also copy the bind files.

Before you can access the DB2 database from your application programs, the BIND files must be bound to the DB2 database in the RHE8 run-time environment. You may use the Setup utility to perform this. The command syntax is as follows:

```
RUN DB2IED model R[ebind] _Path path
```

- Model identify your database.
- \_Path path is used to identify the directory for your program objects.

Note: It is assumed that bind files are placed in sub-directories under your object directory.

Whenever you re-compile application programs and copy these to the RHE8 run-time environment, you must copy the bind files also and use Setup utility top rebind again.

### 3.7 DBSP

DBSP applications are fully supported, however, your applications are executed as direct Oracle applications. The object of DBSP application programs are simply copied to the RHE8 run-time environment, like any other GCOS8 application program. Since no GCOS8/UNIX layers of the application exist, the concepts of pre-connection is slightly different. Refer to DBSP documentation in the development environment.



## 4. TPMS ENVIRONMENT

GCOS8 Rehosting Express use either an embedded TP8 Server or a standard TPMS to host the TP8 Workstation emulation. Following TPMS software is supported:

- BEA Tuxedo 8.1
- IBM TXSeries Encina 5.1
- Micro Focus Enterprise Server 4.0

The TP8 Workstation configuration file replaces the TP8 Workstation Definition on GCOS8. Many of the configuration settings may be copied from the development environment, and amended as required for production execution. If TP8 procedures are, ex. CXPP, CXAP, the object files must be copied from the development environment.

WEB8 and GTEA connections are supported, however, such connections must be done through DSA-Gateway, ref. details under specific TPMS.

### 4.1 TP8 CONFIGURATION

The TP8 Configuration file (\$G8RBDIR\TP8.CFG) is used to define the characteristics of the TP8 Workstation, including the commands accepted at run-time. You should simply copy/paste configuration entries as required from your development environment, and amend these as described below.

#### [CREATE\_WORKSTATION]

WS\_NAME: May be used to change the name of the TP8 Workstation.

#### [CREATE\_TP8\_EXTENSION]

TPR\_POOL: Should be set 0, use PROGRAM-CACHE instead.

PROGRAM\_CACHE : May be used to optimize execution, by caching the most used TPRs and or modules in memory.

STATISTICS: RHE8 will collect statistics for TPR and module use. The maximum setting must be sufficient to cope with the maximum number of TPR and modules within your application. If overflow occur, this is indicated in the Journal and additional statistics are lost.

MAXIMUM\_RESTART : May be used to control how many times a transaction will be restarted in case of concurrency deadlock.

MAXIMUM\_STACK: Whenever a TPR is cancelled from memory (fresh copy), MFE8 has to cancel all modules called by the TPR also to obtain similar behavior as TP8 (RHE8 use dynamic linking). RHE8 will keep track of modules called directly or indirectly through a stack. The maximum setting must be sufficient to cope with the

maximum number of modules called from an application TPR (application modules only).

### **[CREATE\_GLOBAL\_STORAGE]**

Probably the same for development and RHE8 run-time environment.

### **[CREATE\_TX\_STORAGE]**

Probably the same for development and RHE8 run-time environment.

### **[CREATE\_PROFILE]**

Probably the same for development and RHE8 run-time environment.

### **[CREATE\_COMMAND]**

Probably the same for development and RHE8 run-time environment.

The \$RDY command will trigger execution of your ready TPRs. For RHE8 use of ready TPRs are optional. For RHE8 it is not recommended to use ready TPRs (leave out \$RDY), except if these are used to initialize Global Storage(s). All IDS-II databases are opened during TP8 startup, whether or not \$RDY is defined.

### **[CREATE\_SOURCE\_LID]**

Probably the same for development and RHE8 run-time environment.

Note: Connection from any source lid is accepted without configuration, provided the lid is not defined as a destination lid.

### **[CREATE\_DESTINATION\_LID]**

Probably the same for development and RHE8 run-time environment.

### **[CREATE\_TPR]**

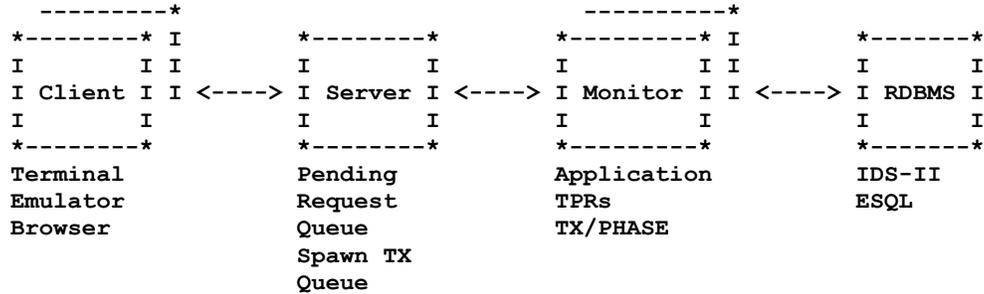
May be used to load high use TPRs and/or modules resident in memory.

## 4.2 TP8 SERVER

The embedded TP8 Server is an emulation of a TP8 Workstation, and use similar techniques - a client connection point (server) and a number of processes (monitor) to for execution of transactions. TP8 Server will scale up to 500 concurrent clients.

The G8RBSDIR environment setting is used for placement of swap files.

### Principles



- Client: Users establish session via TPCLI, DSA-Gateway or Open-Client.
- Server: Messages are queued for processing (PRQ) in Monitor. Output messages are relayed to Client, and Client is given turn when TX completes or goes into conversation.
- Monitor: Executes message from Server - TX or conversational reply.

### TP8 Server

TP8 Server is launched with the RUN TP8S command or icon, and quickly becomes operational, awaiting connections from either clients or monitors. Following configuration entries in Defaults Configuration (G8WB.CFG) may be used to control TP8 Server:

```

[TP-MONITOR-DEFAULTS]
CCI-MODULE      : CCITCP          ; CCIDDE(default) or CCITCP
CCI-PUBLICNAME   : TPMonG8RB      ; Public name of TP8, default
TPMonG8RB
;CCI-TIMEOUT-SECS : 30          ; Default is 120

```

TP8 server communicates with TP8 Monitor(s) via TCP/IP using the name TPMonG8RB.

```

[TP-CLIENT-DEFAULTS]
CCI-MODULE      : CCITCP          ; CCITCP(default) or CCIDDE
CCI-PUBLICNAME   : TPcliG8RB      ; Public name of TP8, default
TPcliG8RB
;CCI-TIMEOUT-SECS : 30          ; Default is 120

```

TP8 server communicates with Client(s) via TCP/IP using the name TPcliG8RB.

Note: If you need to setup multiple RHE8 environments, you need to configure unique public names for Monitor and Client.

```
[TP-SERVER-STARTUP]
;Tpident      Command
;
;           INITIAL and FINAL used to start a process only, all
others
;
;           will start process continuously (start and restart
;           at 30 second interval)
;
;           Note: Tpident must match W8TP ident
;INITIAL      : 'start only once during startup'
;FINAL        : 'start only once during shutdown'
TP1           : 'start <g8rmdir>\bin\RUN TP8M "TP1" /F:\g8rb11'
TP2           : 'start <g8rmdir>\bin\RUN TP8M "TP2" /F:\g8rb12'
```

TP8 Server may be configured to launch one or more Monitors at startup. Should a Monitor stop (RHE8 or uncontrolled application abort), a new Monitor instance is automatically re-launched.

```
[G8WB-TRACE]
TP-SERVER-DEBUG      : OFF
TP-GLOBAL-JOURNAL    : ON
; Trace configuration
; Trace TP Server
; OFF(default)/ON
```

TP8 Server writes statistics to \$g8rmdir\SERVER.LOG.

- TP-SERVER-DEBUG:ON cause a substantial amount of debug/trace information to be added to this log.
- TP-GLOBAL-JOURNAL:ON cause the SERVER.LOG to be appended to \$g8rmdir\SERVER.yyddd file, where yyddd = current julian date.

Following limits are fixed for TP8 Server:

1. Max. number of Clients = 500
2. Max. number of Monitors = 10
3. Min. number of Buffers = 3
4. Max. number of Buffers = 100 (allocated as needed)

RHE8 use an additional environment setting G8RBSDIR for storage of TP8 swap-files. By default G8RBSDIR = G8RBGDIR setting.

## TP8 Server Operation

TP8 Server provides status information in the execution window, and accepts one character commands, as indicated:

```
-> TP Server is in the air
-> Enter Q or S to shutdown TP Server, L for LID Statistics
    M for Monitor Statistics, D for Disconnect Lid
    Z for restart processes, any other for Quick Status
```

Quick Status, which is both shown and written to SERVER.LOG, gives a summary of activity and resource use. Counters are given for Interval (period since last quick status) and Total (period since startup).

```

-> Quick status for TP Server at hh.mm.ss TX-Serial# nnnnnnnn
Counter          Interval /sec    Total /sec
-----
Client  Imsg#  nnnnnnnnn nn.n nnnnnnnnn nn.n
Client  Omsg#  nnnnnnnnn nn.n nnnnnnnnn nn.n
Monitor Imsg#  nnnnnnnnn nn.n nnnnnnnnn nn.n
Monitor Omsg#  nnnnnnnnn nn.n nnnnnnnnn nn.n
Total I/Omsg# nnnnnnnnn nn.n nnnnnnnnn nn.n
Client  Lost#  nnnnnnnnn          nnnnnnnnn
Client  Prq#   nnnnnnnnn          nnnnnnnnn
Client  Max#   nnnnnnnnn          nnnnnnnnn
Resource      Current    Maximum
-----
Clients              nnn          nnn
-active              nnn
-send                nnn
-pending             nnn
-idle                nnn
-spawn               nnn
-sleep               nnn
Monitors             nn          nn
-active              nn
-idle                nn
Buffers              nnn          nnn
-free                nnn

```

Shutdown of TP8 Server may be requested with S or Q command, however, to prevent accidental shutdown, the request has to be confirmed, ex.

```

Shutdown has been requested
Execution is suspended until you reply!
CONFIRM SHUTDOWN (N/Y)?

```

If shutdown is not confirmed with Y (upper-case), the shutdown request is ignored and server.log is copied to server.001 (for viewing). A shutdown will stop all monitors and disconnect all clients.

Note: If you do not have access to the TP8 Server window, you shutdown from a client using the \$SHUTDOWN command.

Shutdown of all idle monitors may be requested with Z command. This feature may be used to reload application programs/libraries without shutting down TP.

Disconnect of a single client may be requested with D command. Current LID statistics is shown, and the LID index is used to designate a specific LID to be disconnected. To prevent accidental shutdown, the request has to be confirmed, ex.

```
-> LID Statistics for TP Server at hh.mm.ss
LID Index Queue   Input# Output# Min. Max. Avg. /sec
-----
xxxx nnn  qqqq   nnnnnnn nnnnnnn nn.n nn.n nn.n

Disconnect of Lid has been requested
Execution is suspended until you reply!
Enter LID Index or just Return
```

If no LID index or an invalid index is given, the disconnect request is ignored.

LID Statistics (L command), which is both shown and written to SERVER.LOG, gives a summary of client activity and current status. Only currently connected clients are shown.

```
-> LID Statistics for TP Server at hh.mm.ss
LID Index Queue   Input# Output# Min. Max. Avg. /sec Lost#
-----
xxxx nnn  qqqq   nnnnnnn nnnnnnn nn.n nn.n nn.n      nnn
      -> Idle, Busy, Send, PRQ, Disc, Spawn, Sleep
```

Note: If unique LIDs are used, then a re-connected client will continue to use the same index (entry), without resetting the counters.

Monitor Statistics (M command), which is both shown and written to SERVER.LOG, gives a summary of Monitor activity and current status. Only currently connected monitors are shown.

```
-> Monitor Statistics for TP Server at hh.mm.ss
Proc Ident   Queue LID      Input#  Output# Min. Max. Avg./sec Start
-----
nn  xxxxxxxx qqqq  nnn-xxxx nnnnnnn nnnnnnn nn.n nn.n nn.n      nnn
      -> Idle, Busy
```

## TP8 Monitor

TP8 Monitor is launched with the RUN TP8M command or icon. Alternatively one or more monitors can be launched by TP8 Server automatically. The command line is used to uniquely identify each monitor. Each monitor must be assigned a non-shared G8RBLDIR. This can be done via command line.

```
RUN TP8M ["TP-Ident"] [ /ldirpath|/. ]
```

- TP-ident: Although optional, a unique identification must be assigned to each monitor, if launched by TP8 Server.
- G8RBLDIR: May be used to set G8RBLDIR for monitor instance.

Following configuration entries in Defaults Configuration (G8WB.CFG) may be used to control TP8 Server:

```
[G8WB-DEFAULTS] ; Defaults configuration
TERMINAL-EMULATION : DKU71XX SERVER ; Emulator [Interface]
; VIP77XX
; VIP78XX
; DKU71XX
; IBM327X
; SERVER
; SERVER TNVIP
; SERVER TN3270
```

The Terminal-emulation setting defines the default terminal emulation for a Client, and SERVER, which instruct the Monitor to run in RHE8 mode. The TNVIP|TN3270 setting defines the default communication mode for the Client interface (TPCLI).

Note: Terminal-emulation can be overridden via Local configuration, however you cannot mix different terminal emulations within the same TP8 server.

```
[TP-MONITOR-DEFAULTS]
CCI-MODULE : CCITCP ; CCIDDE(default) or CCITCP
CCI-PUBLICNAME : TPMonG8RB ; Public name of TP8, default
TPMonG8RB
;CCI-TIMEOUT-SECS : 30 ; Default is 120
```

TP8 Monitor communicates with TP8 Server via TCP/IP using the name TPMonG8RB.

```
[G8WB-TRACE] ; Trace configuration
TP-TX-SEQUENCE : OFF ; Trace TX/TPR state
TP-EXEC-CALL : OFF ; Trace TP Interface/procedure
calls
TP-COMM-DETAIL : OFF ; Trace SEND/RECEIVE details
TP-TPR-EXECUTE : OFF ; OFF(default)/ON
TP-RESPONSE-TIME : ON ; OFF(default)/ON/PROFILE
TP-GLOBAL-JOURNAL : ON ; OFF(default)/ON
```

- TP-RESPONSE-TIME:ON cause a one line trace to be written to \$g8rbldir\ JOURNAL.LOG for each TX/PHASE.
- TP-GLOBAL-JOURNAL:ON cause the JOURNAL.LOG to be appended to \$g8rbgdir\JOURNAL.yyddd file, where yyddd = current julian date.
- TP-TPR-EXECUTE should be left OFF, otherwise a message is sent to the client, whenever a TPR is executed. This message will, however, not be shown at the client.
- TP-TX-SEQUENCE, TP-EXEC-CALL, TP-COMM-DETAIL:ON cause a substantial amount of debug/trace information to be written to JOURNAL.LOG.

Note: Monitor(s) will continue to run until TP8 Server is closed down, or a time-out occur while waiting for TP8 Server to respond.

## TP8 Client

TP8 Clients may be connected either through TP8 Client utility (TPCLI), through a DSA-Gateway or through Open-client.

### TP8 Client Utility

TP8 Client is launched with the RUN TPCLI command or icon. The optional command line may be used to control client execution mode and override default configuration. Following modes are possible:

1. Run a single TX
2. Run a series of TX's from a file
3. Start a terminal dialog using embedded emulators
4. Start a terminal dialog via an external terminal emulator

The syntax of command line is as follows:

```
RUN TPCLI [BIBOTX "command"|BIBOFILE bibo-file [think-secs]]  
          [lid] [terminal-emulation]  
          [/ldirpath|/.]
```

- BIBOTX - may be used to execute a single TP8 message - given in "command", and then terminate.
- BIBOFILE - may be used to execute a series of TP8 messages from a bibo-file. Messages in the file can either be plain text or in BIBO format (see MFE8 documentation for details). The optional think-secs may be used to slow down execution of messages.

If neither BIBOTX nor BIBOFILE is specified, a terminal dialog is started.

- Lid - may be used to set a lid for the session, and thus override default configuration.
- Terminal-emulation - may be used to set terminal emulation and interface for the session, and thus override default configuration.

```
VIP77XX|VIP78XX|DKU71XX|IBM327X|TTY SERVER [TNVIP|TN3270]
```

Note: SERVER must be specified for RHE8. TNVIP or TN3270 trigger client utility to acts as an interface to terminal emulation software.

- G8RBLDIR: The setting for G8RBLDIR may be changed on-the-fly, ex.

```
/c:\temp (specific path)  
/. (current path)
```

Note: G8RBLDIR is ONLY used for optional local configuration, no files, except for Hard-copy, are written.

Following configuration entries in Defaults Configuration (G8WB.CFG) may be used to control TPCLI:

```
[TP-CLIENT-DEFAULTS]
CCI-MODULE          : CCITCP          ; CCITCP(default) or CCIDDE
CCI-PUBLICNAME       : TPcliG8RB      ; Public name of TP8, default
TPcliG8RB
;CCI-TIMEOUT-SECS   : 30              ; Default is 120
```

TP8 Client communicates with TP8 Server via TCP/IP using the name TPcliG8RB.

```
[G8WB-DEFAULTS]
LOGICAL-ID          : MAST            ; Defaults configuration
; Default LID for TP Log-on
```

Note: It is recommended to use unique LIDs for TP8 Clients, as this makes it possible to identify a user and collect useful statistics.

### TP8 Client DSA-Gateway

The DSA-Gateway is based on Ggate from G&R, and you need both Ggate and GI-API run-time licenses in order to use DSA-Gateway. With DSA-Gateway you are able to connect Glink to TP8 Server, as if TP8 Server was a GCOS8 node. The setup, configuration and use of DSA-Gateway is documented in G8RB, ref. Open-Client-Toolkit.

For RHE8 the 'Listener' configuration of Ggate is slightly different and only ONE entry is required.

```
listen -mn rhftp8 -ext -lim nn -cmd <command>
```

- 'rhftp8' is the mailbox to listen and handle connection for
- nn designate the maximum number of connections.

The <command> syntax is:

```
W8GLAPI [options] publicname -ln -mn mailbox -li dsa [gline args]
```

- 'W8GLAPI' is the program launched when connection is detected.
- 'publicname' must be the same as CC-PUBLICNAME:<name> as configured under [TP-CLIENT-DEFAULTS].
- 'options' can be VIP77XX|VIP78XX|DKU71XX|IBM327X|TTY|BROWSER, and designates the terminal-emulation for the session. The default is defined in Default Configuration (G8WB.CFG).

Note: For IBM327X emulation you MUST use W8GLAPIE program, which implies IBM327X emulation.

- 'options' can be DEBUG|DEBUGALL, where DEBUG may be used to create a debug file under the 'servers' directory of Ggate. DEBUGALL may be used to create a new debug file for each session.
- 'options' can be "tx", may be used to trigger a logon transaction when a session is established, ex. "VLOGO" will run the tx = VLOGO without any further input.

- 'options' can be NODIALOG, may be used to suppress the connect/hello messages from TP8.
- 'options' can be NOETX, may be used to suppress ETX, which is otherwise appended to messages sent to/from TP8.
- 'mailbox' must be the same as specified in '-mn mailbox', ex. rhtp8.
- 'gline args' may optionally be specified, ex. -d\_ will trigger Gline debug file creation.

Here's a full example (all on one line):

```
listen -mn rhtp8 -ext -lim 50 -cmd w8glapi TPcliG8RB dku71xx
    -ln -mn rhbtp8 -li dsa -t_ -s_
```

Here's an example of a connection through Glink:

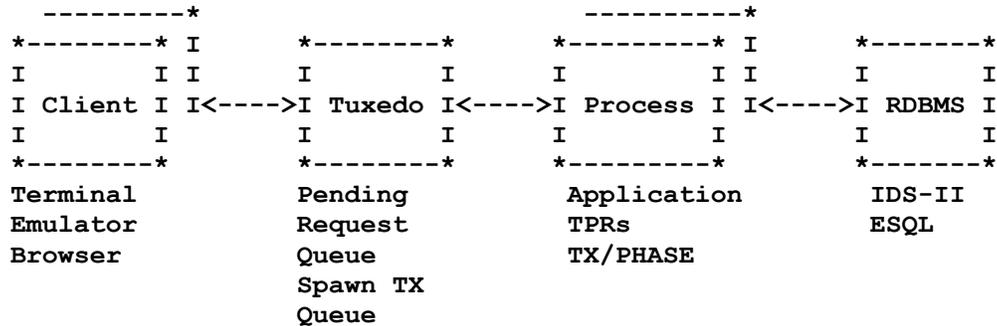
1. Communication interface = Windows Sockets
2. IP address/Host = Ggate Server
3. Protocol = G&R DSA Gateway
4. Host = myggate or ip
5. Host mailbox = RHETP8
6. Host node = GGATE\_RFC
7. Mailbox ext = LID1 (optional Lid, but recommended)

Note: Terminal emulation setting in Glink MUST match the terminal emulation used for the session.

## 4.3 BEA TUXEDO

TUXEDO/TP8 is an emulation of a TP8 Workstation within the standard TUXEDO TPMS. TUXEDO/TP8 will well scale beyond the 500 concurrent client limit of TP8 Server, and integrate the GCOS8 application into a standard TUXEDO execution environment.

### Principles



- Client: Users establish session via TPCLI or DSA-Gateway, which includes interface to Tuxedo Workstation.
- Tuxedo: Messages are queued for processing in Tuxedo. Output messages are relayed to Client, and Client is given turn when TX completes or goes into conversation.
- Process: Executes message from Tuxedo - TX or conversational reply.

### Tuxedo Workstation

TP8 handling must be defined in the Tuxedo configuration file, you may select either of three Tuxedo servers to execute TP8 commands, as follows:

- TUXTP8 - May be used to execute TP8 messages (conversational or non-conversational). Database commit/rollback is controlled by RDBMS software.
- TUXTP8TX - Like TUXTP8, except TP8 messages are executed under Tuxedo transaction control. Should be used if application is spawning transaction(s), and you wish to ensure rollback of spawn operations, if TX fails.
- TUXTP8XA - Like TUXTP8TX, except database commit/rollback are coordinated with Tuxedo using the XA protocol. Should be used if application is amended to start of sub-transactions, which require coordinated commit/rollback.

The \$g8rbgdir\tuxedo\rhe8.cfg and rhe8w.cfg files provides examples of Tuxedo configuration files, and you can either use these to create a new Tuxedo workstation or amend an existing one.

Example (simple):

```

*GROUPS
GROUP2
  LMID=simple      GRPNO=2  OPENINFO=NONE
  
```

```

*SERVERS
DEFAULT:
    CLOPT="-A"
TUXTP8    SRVGRP=GROUP2 SRVID=1
          MIN=1 MAX=2
TUXTP8TX  SRVGRP=GROUP2 SRVID=5001
          MIN=1 MAX=2
*SERVICES
TUXTP8
TUXTP8TX

```

Above defines the two servers TUXTP8 and TUXTP8TX, and associates the with groups, which trigger spawn of processes to handle TP8 messages.

When Tuxedo is launched with 'tmboot' command, the 'min' numbers of processed are launched for each server, ready for processing of TP8 messages. Additional processes are launched as required up to the 'max' number specified.

Tuxedo is stopped with 'tmshutdown' command.

Example (/WS): If Tuxedo client are not executed on the same server as Tuxedo workstation, you also need to configure a /WS listener (rhe8w.cfg).

```

*MACHINES
MYSERVER    LMID=simple
            MAXWSCLIENTS=100
*GROUPS
GROUP1
            LMID=simple    GRPNO=1 OPENINFO=NONE
*SERVERS
WSL        SRVGRP=GROUP1 SRVID=1
            CLOPT=" -- -n //myserver:25000"
            RQPERM=0660    REPLYQ=Y        RPPERM=0660
            MIN=1          MAX=1          CONV=N
            SYSTEM_ACCESS=FASTPATH
            MAXGEN=5       GRACE=86400    RESTART=Y

```

Example (Spawn): If the TP8 application is spawning transaction (other than SWTM to originator), or send to ROP lids, you also need to configure one or more queues.

```

*GROUPS
GROUP4
            LMID=simple GRPNO=4 TMSNAME=TMS_QM TMSCOUNT=2
            OPENINFO="TUXEDO/QM:<path>\QUE;QSPACE"
*SERVERS
TMQUEUE    SRVGRP=GROUP4 SRVID=1
            GRACE=0 RESTART=Y CONV=N MAXGEN=10
            CLOPT="-s QSPACE:TMQUEUE"

```

Above creates a minimum queue mechanism for handling of TX spawn and/or ROP messages. Before starting Tuxedo, you must use the 'qmadmin' command as follows:

1. Use 'crdl' directive to create the device
2. Use 'qspacecreate' to create a qspace for storage
3. Use 'qcreate' to create on or more queues within qspace

For TX spawn you must create a queue with the name "TP8<slid>" for each 'spawn-lid' defined in your TP8 Workstation configuration.

For ROP messages you must create ONE queue with the name 'TP8ROP'. Please refer to Tuxedo /Q for details on queue administration.

Example (XA): If TP8 application is to be executed as part of a distributed transaction, i.e. TX is launching sub-transaction(s) which required coordinated commit/rollback, you need to configure the TUXTP8XA server and an associated resource manager.

```
*GROUPS
UDB_GRP
    LMID=simple      GRPNO=3
    TMSNAME=TMS_UDB TMSCOUNT=2
    OPENINFO="UDB_XA:db=MYDB,uid=db2admin,pwd=xxx,tpm=TUXEDO"
*SERVERS
TUXTP8XA      SRVGRP=UDB_GRP SRVID=10001
              MIN=1 MAX=2
*SERVICES
TUXTP8XA
```

Above example demonstrates database access through DB2 UDB. During startup of TUXTP8, TUXTP8TX and TUXTP8XA processes, the TP8 Workstation is configured through G8WB.CFG and TP8.CFG configuration files, however this configuration takes place ONLY for the first process, all others processes will share the same configuration, until Tuxedo is shutdown and re-booted.

Although Tuxedo treat all TP8 messages as non-conversational, the RHE8 emulation of TP8 will keep track of the session state, and handle checkpoint, conversations and abort like TP8. The state information (TP-Storage, TX-Storage, Constant-Storage and Lid-Storage) is swapped as required during TP8 execution.

RHE8 use an additional environment setting G8RBSDIR for storage of TP8 swap-files. By default G8RBSDIR = G8RBGDIR setting.

## TP8 Configuration

RHE8 will log a few information messages, and any fatal exception messages to the standard Tuxedo log. In addition a TP8 Journal is used to provide further details of events and exception occurring during TP8 execution. The TP8 Journal is placed under \$G8RBLDIR\journal.<pid>.log, where pid = process-id of the Tuxedo process.

Following configuration entries in Defaults Configuration (G8WB.CFG) may be used to TP8 journal content:

```
[G8WB-TRACE]                ; Trace configuration
TP-TX-SEQUENCE      : OFF    ; Trace TX/TPR state
```

```

TP-EXEC-CALL      : OFF          ; Trace TP Interface/procedure
calls
TP-COMM-DETAIL   : OFF          ; Trace SEND/RECEIVE details
TP-TPR-EXECUTE   : OFF          ; OFF(default)/ON
TP-RESPONSE-TIME : ON           ; OFF(default)/ON/PROFILE
TP-GLOBAL-JOURNAL : ON          ; OFF(default)/ON

```

- TP-RESPONSE-TIME:ON cause a one line trace to be written to journal for each TX/PHASE.
- TP-TPR-EXECUTE should be left OFF, otherwise a message is sent to the client, whenever a TPR is executed. This message will, however, not be shown at the client.
- TP-TX-SEQUENCE, TP-EXEC-CALL, TP-COMM-DETAIL:ON cause a substantial amount of debug/trace information to be written to journal.
- TP-GLOBAL-JOURNAL:ON cause the journal to be appended to \$g8rbgdir\JOURNAL.yyddd file, where yyddd = current julian date, upon completion of process.

Following configuration entries in TP8 Configuration (TP8.CFG) may be used to control Tuxedo behavior:

```

[create_monitor_extension]
ROP                : IGNORE      ; IGNORE(default)|QUEUE
SLAVE              : LOG         ; LOG(default)|IGNORE|QUEUE
SPAWN              : IGNORE      ; IGNORE(default)|QUEUE
QUEUE_SPACE_NAME  : QSPACE      ; QSPACE(default)|<qspacename>

```

- ROP, QUEUE will cause messages to ROP lids to be queued in Tuxedo for further processing, otherwise such messages are ignored.
- SPAWN, QUEUE will cause messages to SPAWN lids to be queued in Tuxedo for further processing, otherwise such messages are ignored.
- SLAVE, QUEUE will cause messages to SLAVE lid to be queued in Tuxedo for further processing (the queue name is TP8SLAVE). LOG will cause messages to SLAVE lid to be written to the Tuxedo Log, otherwise such messages are ignored.
- QUEUE\_SPACE\_NAME setting may be used to change the default QSPACE name for the Tuxedo Queue Space.

## TP8 Client

TP8 Clients may be connected either through Tuxedo/TP8 Client utility (TP8TUX), through a DSA-Gateway or through another Tuxedo.

### Tuxedo/TP8 Client Utility

Tuxedo/TP8 Client is launched with the TP8TUX or TP8TUXW command or icon(s).

TP8TUX can be used when executed on the same server as the Tuxedo Workstation. The environment variable TUXCONFIG must point to the configuration file for Tuxedo Workstation, ex. c:/mytux/tuxconfig.

TP8TUXW must be used when executed elsewhere (connect through /WS). The environment variable WSNADDR must point to the listening address of the Tuxedo Workstation, ex. //myserver:25000.

The optional command line may be used to control client execution mode. Following modes are possible:

1. Run a single TX
2. Run a series of TX's from a file
3. Start a terminal dialog using embedded emulators
4. Start a terminal dialog via an external terminal emulator

The syntax of command line is as follows:

```
TP8TUX[W] [BIBOTX "command"|BIBOFILE bibo-file [think-secs]]
          [TUXTP8|TUXTP8TX|TUXTP8XA]
          [/[username:[clientname:]]password]
          [lid] [terminal-emulation]
```

- BIBOTX - may be used to execute a single TP8 message - given in "command", and then terminate.
- BIBOFILE - may be used to execute a series of TP8 messages from a bibo-file. Messages in the file can either be plain text or in BIBO format (see MFE8 documentation for details). The optional think-secs may be used to slow down execution of messages.

- If neither BIBOTX nor BIBOFILE is specified, a terminal dialog is started.
- TUXTP8|TUXTP8TX|TUXTP8XA - may be used to specify the Tuxedo service. The default is TUXTP8.
- Lid - may be used to set a specific lid for the session.
- /... - may be used to provide authentication information for Tuxedo, if configured. For SYSAUTH only password is required. If authentication is configured and no information is given, you are prompted for information.
- Terminal-emulation - may be used to set terminal emulation and interface for the session, and thus override default configuration.

```
VIP77XX|VIP78XX|DKU71XX|IBM327X|TTY [TNVIP|TN3270]
```

Note: TNVIP or TN3270 trigger client utility to acts as an interface to terminal-emulation software, ex. Glink.

Note: It is recommended to use unique LIDs for TP8 Clients, as this makes it possible to identify a user and collect useful statistics.

### **Tuxedo/TP8 Client DSA-Gateway**

The DSA-Gateway is based on Ggate from G&R, and you need both Ggate and GIAPI run-time licenses in order to use DSA-Gateway. With DSA-Gateway you are able to connect Glink to Tuxedo/TP8, as if Tuxedo/TP8 was a GCOS8 node. The setup, configuration and use of DSA-Gateway is documented in G8RB, ref. Open-Client-Toolkit.

For RHE8 the 'Listener' configuration of Ggate is slightly different and only ONE entry (per Tuxedo service) is required.

```
listen -mn tuxtp8 -ext -lim nn -cmd <command>
```

- 'tuxtp8' is the mailbox to listen and handle connection for
- nn designate the maximum number of connections.

The <command> syntax is:

```
TUXGLAPI[W] [options] -ln -mn mailbox -li dsa [gline args]
```

- 'TUXGLAPI' is the program launched when connection is detected and can be used when Ggate is executed on the same server as the TuxedoWorkstation. The environment variable TUXCONFIG must point to the configuration file for Tuxedo Workstation, ex. c:/mytux/tuxconfig.
- 'TUXGLAPIW' is the program launched when connection is detected and must be used when Ggate is executed elsewhere (connect through /WS). The environment variable WSNADDR must point to the listening address of the Tuxedo Workstation, ex. //myserver:25000.
- 'options' can be TUXTP8|TUXTP8TX|TUXTP8XA, and this designates the Tuxedo service to use for the session. The default is TUXTP8.
- 'options' can be VIP77XX|VIP78XX|DKU71XX|IBM327X|TTY|BROWSER, and designates the terminal-emulation for the session. The default is defined in Default Configuration (G8WB.CFG).

Note: For IBM327X emulation you MUST use TUXGLAPIE or TUXGLAPIEW program, which implies IBM327X emulation.

- 'options' can be DEBUG|DEBUGALL, where DEBUG may be used to create a debug file under the 'servers' directory of Ggate. DEBUGALL may be used create a new debug file for each session.
- 'options' can be "tx", may be used to trigger a logon transaction when a session is established, ex. "VLOGO" will run the tx = VLOGO without any further input.
- 'options' can be NODIALOG, may be used to suppress the connect/hello messages from TP8.
- 'options' can be NOETX, may be used to suppress ETX, which is otherwise appended to messages sent to/from TP8.
- 'mailbox' must be the same as specified in '-mn mailbox', ex. tuxtp8.
- 'gline args' may optionally be specified, ex. -d\_ will trigger Gline debug file creation.

Here's a full example (all on one line):

```
listen -mn tuxtp8 -ext -lim 50 -cmd tuxglapiw tuxtp8 dku71xx  
-ln -mn tuxtp8 -li dsa -t_ -s_
```

Here's an example of a connection through Glink:

1. Communication interface = Windows Sockets
2. IP address/Host = Ggate Server
3. Protocol = G&R DSA Gateway

4. Host = myggate or ip
5. Host mailbox = TUXTP8
6. Host node = GGATE\_RFC
7. Mailbox ext = LID1 (optional Lid, but recommended)

Note: Terminal emulation setting in Glink MUST match the terminal emulation used for the session.

If Tuxedo is configured for authentication - SYSAUTH or APPAUTH, you must setup Glink to provide following information (fixed or prompted):

- password - minimum for SYSAUTH
- userid - used as username for Tuxedo
- billing - used as clientname for Tuxedo

### **Tuxedo Queue Utility**

If messages are queued from Tuxedo/TP8 - spawn, ROP or SLAVE, you must start one or more Tuxedo Queue utilities to process the queued messages. Tuxedo Queue Utility is launched with the QUETUX or QUETUXW command or icon(s).

- QUETUX can be used when executed on the same server as the Tuxedo Workstation. The environment variable TUXCONFIG must point to the configuration file for Tuxedo Workstation, ex. c:/mytux/tuxconfig.
- QUETUXW must be used when executed elsewhere (connect through /WS). The environment variable WSNADDR must point to the listening address of the Tuxedo Workstation, ex. //myserver:25000.

The optional command line may be used to control queue execution mode. The syntax of command line is as follows:

```

QUETUX[W] [SLAVE programname]
           [ROP programname]
           [SPAWN [[lid|group]...]]

           Default is SPAWN (all lids)

```

- SLAVE - send SLAVE messages to the specified programname
- ROP - send ROP messages to the specified programname
- SPAWN - execute spawned messages for the specified lid(s) and or group(s). Default is all lids.

Note: Timed messages are handled by /Q.

In the command window you are able to stop QUETUX using Q or S key.

## **4.4 IBM TXSERIES ENCINA**

Under construction ...

## **4.3 MICRO FOCUS ENTERPRISE SERVER**

Under construction ...

## INDEX

Application Server .....	16	Micro Focus Enterprise Server .....	73
BEA Tuxedo .....	65	Microsoft SQL-Server RDBMS .....	50
Copyright Information .....	12	Oracle RDBMS .....	40, 53
Coverpage .....	1	Preface .....	7
DBSP .....	53	RDBMS .....	29
Development environment.....	17	Related Documentation .....	11, 12
Environment .....	20	RHE8 Installation .....	13
IBM DB2 RDBMS .....	45	Running batch programs .....	19
IBM TXSeries Encina.....	72	Running utilities .....	23
IDS-II configuration .....	31	Server Installation.....	13
Informix RDBMS .....	33	Single-Platform Installation .....	16
Installation information .....	13	TP8 configuration .....	55
Integration.....	16, 17	TP8 Server .....	57
Interel/RFM.....	51	TPMS .....	55
Introduction .....	9	Workstation Installation .....	14
JCL/Job streams .....	26		