The Complete
Sybase Replication Server
Quick Reference Guide

Versions 15.1, 15.2, 15.5, 15.6, 15.7 & 15.7.1
3rd edition

These pages are taken from "The Complete Sybase Replication Server Quick Reference Guide", 3rd edition, as a sample of the contents. Note that the original page size is 11 by 22 cm (4.3 by 8.6 inch).
The complete book can be ordered from http://www.sypron.nl/repqr.

Copyright © 2004-2012 Sypron B.V

Rob Verschoor

Sypron Publications

ISBN 978-90-806117-3-3
The Complete Sybase Replication Server
Quick Reference Guide
Replication Server versions 15.1, 15.2, 15.5, 15.6, 15.7 & 15.7.1
3rd edition

by Rob Verschoor

ISBN 978-90-806117-3-3

Published by:
Sypron B.V.
Het Wolfseind 24
3823 VS Amersfoort
The Netherlands
Internet www.sypron.nl
Email sypron@sypron.nl

Printed in The Netherlands

First edition : July 2004 (RS 12.0, 12.1, 12.5 & 12.6)
Second edition : June 2008 (RS 12.6, 15.0.1 & 15.1)
Third edition : July 2012 (RS 15.1, 15.2, 15.5, 15.6, 15.7 & 15.7.1)

Sybase is a registered trademark of Sypron B.V.
Sybase, Transact-SQL, Adaptive Server Enterprise, Replication Server, Replication Agent, Adaptive Server Anywhere, SQL Anywhere, OpenServer, OpenSwitch, Rep-Connector, OmniConnect, DirectConnect, EnterpriseConnect, Enterprise Connect Data Access and ExpressConnect (among others) are registered trademarks of Sybase, Inc. Other product or brand names may be (registered) trademarks of their respective owners.

Copyright © 2004-2012 Sypron B.V. All rights reserved. No part of this publication may be reproduced in any form, or by any means, without the prior written permission of the publisher.

While this book has been prepared with care, neither the author, nor the publisher, nor Sybase Inc. (nor its subsidiaries or parent company), assume any responsibility for errors or omissions, nor do they accept any liabilities for damages resulting from the use of the information herein.
Contents

Introduction .................................................................................................................. 6
Replication Server versions covered in this edition ................................................. 6
How complete is 'Complete'? (topics not covered) .............................................. 6
How to use this Quick Reference Guide ............................................................. 7
Syntax conventions ................................................................................................... 7
Terminology and Acronyms ...................................................................................... 7
Errata ............................................................................................................................. 8
Electronic Supplement ............................................................................................ 8
Undocumented commands: warning & disclaimer .................................................. 8

Overview & General Concepts ............................................................................. 9
1 Starting points for common RepServer DBA tasks ............................................ 9
2 RepServer basics ................................................................................................... 9
  2.1 Data model considerations for replication systems ....................................... 10
  2.2 Why replication systems can be difficult....................................................... 11
3 Sybase data replication products ..................................................................... 12
  4 Products required for heterogeneous replication ............................................ 12
  5 License keys & editions ...................................................................................... 13
  5.1 RepServer Editions ....................................................................................... 14
6 RepServer documentation ................................................................................... 14

RepAgents & Primary Databases ....................................................................... 15
7 Marking tables/procedures/databases for replication .......................................... 15
  7.1 Primary ASE databases .................................................................................. 15
  7.2 Marking data for warm standby or database replication .......................... 16
  7.3 Be careful when marking big text/image tables in ASE! ......................... 18
  7.4 Using SQL statement replication (RS 15.2, ASE 15.3, 15.4) .................... 18
  7.5 ASE secondary truncation point vs. RSSD locator ..................................... 19
  7.6 Replicating to/from ASE In-Memory databases (15.6) ............................ 20
  7.7 Primary non-ASE databases ....................................................................... 20
8 Replication Agents .............................................................................................. 20
  8.1 ASE RepAgent configuration ...................................................................... 21
  8.2 ASE RepAgent troubleshooting (undocumented) ...................................... 23
  8.3 Non-ASE RepAgents ................................................................................... 24

Core Replication Functionality ....................................................................... 26
9 RepServer syntax basics ...................................................................................... 26
10 Getting RepServer status information .............................................................. 26
11 Datatypes, datatype classes & -translations ..................................................... 27
  11.1 Specifying datatypes in a repdef ................................................................. 28
  11.2 Datatype classes and datatype translations .............................................. 28
  11.3 RSSD tables - datatypes .......................................................................... 29
12 Connections to RepAgents ............................................................................... 29
13 Connections to primary/replicate ASE databases .......................................... 30
  13.1 RSSD tables - database connections ......................................................... 32
  13.2 Getting information about database connections ................................... 33
  13.3 Configuration parameters for database connections .............................. 33
14 Connection Profiles for non-ASE databases (15.2) ....................................... 38
  14.1 RSSD tables - connection profiles .............................................................. 39
15 Bulk replication with HVARRTL (15.5) ............................................................ 39
  15.2 Replicating into Sybase IQ with RTL (15.5) ................................................. 40
  15.3 Configuration parameters for HVAR/RTL (15.5) .................................... 41
16 Replication definitions ......................................................................................... 42
  16.1 Table repdefs ............................................................................................. 42
  16.2 Schema changes for replicated tables (15.5) .......................................... 43
  16.3 Schema changes for replicated tables (pre-15.5) ....................................... 46
  16.4 Function repdefs (stored procedure replication) ....................................... 47
  16.5 Replication functions & Request functions .............................................. 49
  16.6 Database repdefs (MSA) ......................................................................... 51
  16.7 Replicating database repdefs with database repdefs/subs .................... 54
  16.9 Warm standby vs. Database repdefs/subscriptions ................................ 55
  16.10 Replicating data in the ASE 'master' database ....................................... 55
  16.11 'Standby' option in repdefs ................................................................. 56
  16.12 Replicating minimal columns ............................................................... 56
  16.13 Mixing table replication and procedure replication ............................... 57
  16.14 Mixing database repdefs and table/function repdefs ........................... 58
  16.15 Using table repdefs in warm standby, DB replication ......................... 59
  16.16 RSSD tables - repdefs ............................................................................ 59
  16.17 Getting information about repdefs ............................................................ 59
17 Replicating text/image/rawobject/LOB columns ........................................... 59
Introduction

Sybase Replication Server was designed as a versatile and powerful software product, and that goal has certainly been achieved. Unfortunately, it seems as if RepServer was never meant to be easy to use -- some of the RepServer syntax isn't easy to remember, and troubleshooting may require you to look for clues in a dozen different places. As a result, a DBA can quickly find him/herself at the wrong side of all that power and versatility.

To be fair, some of these issues can hardly be avoided: replication systems are never simple, and will quickly be an order of magnitude more complex than a single database. Nevertheless, many infrequently used commands and concepts of RepServer tend to slip your mind, especially when your replication system keeps running without any problems -- which, fortunately, is the case most of the time.

Invariably though, the moment comes when you need to take action, and that's when this book will be convenient: it contains all RepServer commands, RSSD procedures and ASE procedures you might need in practice. Apart from describing the basics of each command, I have also included some background information and practically useful tips, as well as a few handy RSSD queries for some of those things you sometimes need to figure out (well, at least I had to…).

When you're a RepServer DBA, there are those moments when you'll need all the help you can get -- this book will be your companion.

I'd like to thank everyone who bought previous editions of this quick reference guide and/or provided input and feedback.

Most of all, my gratitude goes to Simone for her invaluable support and understanding while I was busy writing this book.

Rob Verschoor
Amersfoort, July 2012, 3rd edition

About this Quick Reference Guide

Replication Server versions covered in this edition

The information in this Quick Reference Guide applies to RepServer version 15.1 or later, a Unix/Linux environment, except where indicated otherwise.

Most of this book assumes ASE-to-ASE replication for ASE version 15.0 or later, although most commands and features apply just as well to non-ASE replication or earlier ASE version, though this may not be indicated explicitly.

Briefly, the following RepServer versions are covered in this book:

- **15.1** - the base RepServer version used for this book.
- **15.2** - main new features: SQL statement replication, and connection profiles.
- **15.5** - with the introduction of the new features of HVAR and RTL, RepServer requires significantly more memory to cache replicated data. Therefore, RepServer is now primarily available in 64-bit (32-bit versions are being phased out). Use `admin version` to determine if your RepServer is 32- or 64-bit.
- **15.6** - added support for RTL from Oracle to IQ.
- **15.7** - main new features: multi-path replication (MPR) for fundamentally higher replication throughput.
- **15.7.1** - added support for MPR from/to Oracle and to IQ; also introduced new password security features.

Much of the information in this book is still valid for pre-15.1 RepServer versions, but this may not be indicated explicitly.

Throughout this Quick Reference Guide, notations like '(15.2)' indicate new or changed functionality in the corresponding RepServer version (here, 15.2); likewise, '(pre-15.2)' marks functionality that existed until version 15.1, but not in 15.2 or later. Unless indicated otherwise, all version numbers refer to RepServer. When versions of other products are involved, this is indicated explicitly, for example 'ASE 15.5 and RS 15.7'. Here, 'RS' indicates RepServer to avoid any confusion.

How complete is 'Complete'? (topics not covered)

I have called this RepServer Quick Reference Guide 'Complete', because it covers almost all aspects of RepServer you may need in daily practice -- although any book about a software product as complex as RepServer must be incomplete by definition. Also, it is obvious that a 160-page booklet simply cannot contain the same amount of information as a bookshelf filled with full-size manuals -- no matter how small you print. For practical reasons, the following topics have therefore been omitted:

- System design aspects are not covered (except for some basic data modelling
aspects; p.10). Although essential for any replication system, a quick reference guide is not the best place for discussing system design issues. See the Replication Server Design Guide (p.14) for details.

- Only some aspects of heterogeneous replication are covered, such as a selection of the commands for non-ASE RepAgents (p.24).
- Replication system recovery is not included, apart from mentioning it briefly (p.130). There is simply too much to say about this rather complex topic, and including a summary would only create a false impression of completeness. Instead, see the Replication Server Administration Guide and Replication Server Troubleshooting Guide (p.14).
- Some features are covered only partly, or not at all, for example LDAP, SSL, security mechanisms and High-Availability failover for the RSSD.
- Although most actions can also be performed via Sybase Control Center or Sybase Central, this book is primarily focused on the actual RepServer commands and procedures, as executed manually through isql.

Note that this Quick Reference Guide is not a tutorial; the reader is expected to be familiar with the basic concepts of RepServer. Nevertheless, if you're new to RepServer, you may find it useful to read section 2 (p.9); also, Section 1 contains some starting points for common DBA tasks.

How to use this Quick Reference Guide
All information in this Quick Reference Guide is grouped by topic, such as 'Table repdefs' or 'Routes' (see the table of contents; p.3). The index (p.147) contains all commands, functions, stored procedures, etc., in alphabetical order. Many references are included in the index more than once; for example, rs helppub is also included as helppub, rs_.

Each command description includes the things you need most: the syntax, a short description of the functionality and parameters, and often an example. This information will usually be sufficient; when you need more details or background information, see one of the RepServer product manuals, like the Replication Server Reference Manual or the Replication Server Administration Guide (p.14).

Syntax conventions
The syntax notation follows commonly used conventions, and will usually be self-explanatory. Still, here are some guidelines:

- Literal text, such as command keywords, is printed in bold. Placeholders for variables or parameters are printed in italic—you should replace this with something. For example: drop replication definition repdef_name.
- Parentheses -(‘ and ‘) and commas are part of the command itself. Square brackets -‘[‘ and ‘]’-, curly brackets -‘{‘ and ‘}’-, pipes -‘|’-, and 3-dots -‘…’- are never part of a command, but indicate optional parts of the command.
- In some cases, the syntax has been simplified for practical reasons, omitting non-essential parts for better readability. For example [ { NULL | table_name } ] is used, although [ { NULL | table_name } ] would have been more formally correct.

Terminology and Acronyms
The following terminology and acronyms are used in this Quick Reference Guide:

RS, RepServer Replication Server
RepAgent Replication Agent
LTM Log Transfer Manager (an old name for RepAgent)
DSI Data Server Interface, an RS thread connecting to a replicate database
RSI Replication Server Interface, an RS thread connecting to another RepServer for a route
RCL Replication Server Command Language; this acronym is used in the Sybase documentation, but not in this book
command a RepServer language command (alter connection)
statement an SQL language statement (select, create table)
repdef replication definition
route a one-way connection between two RepServers
dataserver generic name for a database server, either ASE or non-ASE, containing or managing a database
primary dataserver/database a dataserver/database containing primary data
replicate dataserver/database a dataserver/database containing replicate data
primary table/procedure a table/stored procedure, located in a primary database, which is the source of replicated data
replicate table/procedure

primary RepServer

replicate RepServer

RSSD

ERSSD

primary RSSD

replicate RSSD

.cfg file

warm standby pair

database replication

quiescing

replication domain

heterogeneous replication

bi-directional replication

system [stored] procedures

ASE

SQLA

ASA

DDL

DML

LTL

OQID

T-SQL

bcp

OS

a.k.a.

Errata

Electronic Supplement

Undocumented commands: warning & disclaimer
# Overview & General Concepts

## 1 Starting points for common RepServer DBA tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Recommended action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a new RepServer</td>
<td>Use <code>rs_init</code> (p.108)</td>
</tr>
<tr>
<td>Start or stop a RepServer</td>
<td>See section 31 (p.108)</td>
</tr>
<tr>
<td>Determine the current status of a RepServer</td>
<td>Use RepServer commands like <code>admin who</code> (p.26)</td>
</tr>
<tr>
<td>Determine the RSSD for a RepServer</td>
<td>Run <code>admin rssd_name</code> and/or <code>sysadmin erssd</code>. See section 39 (p.121)</td>
</tr>
<tr>
<td>Run an RSSD stored procedure (like <code>rs_helprep</code>)</td>
<td>First, connect to the RSSD dataserver with <code>isql</code> (p.110); then run the RSSD procedure</td>
</tr>
<tr>
<td>Create a connection to a primary or replicate database</td>
<td>Use <code>rs_init</code> (p.108)</td>
</tr>
</tbody>
</table>
| Set up publish-subscribe replication for tables or stored procedures | 1. Create connections to a primary and a replicate database (see above)  
2. Create a repdef for a primary table or stored procedure (p.42, 47). Multiple repdefs can also be grouped in a publication (p.62)  
3. Mark the primary table or stored procedure for replication (p.15)  
4. Create one or more subscriptions to the database repdef (p.64) |
| Set up a warm standby | See section 22.5 (p.75) |
| Set up database replication | 1. Create connections to a primary and replicate database (see above)  
2. Create a database repdef (p.51)  
3. Mark the database for replication (p.15)  
4. Create one or more subscriptions to the database repdef (p.64) |
| Enable password encryption for a RepServer | Use `rs_init` (p.108) |
| Reverse-engineering of RepServer objects | Use RMP, RSP or PowerDesigner (p.141) |
| Change configuration parameters | See section 27 (p.99) |
| Find out why replication isn't working | See section 50 (p.136) |
| Improve performance of your replication system | See section 47 (p.131) |
| Set up an ASE-to-ASE or Oracle-to-Oracle replication reference environment | (15.5) Follow the instructions in the Replication Server Administration Guide, Vol. 2, appendix Implement a Reference Replication Environment |

If you need to create RepServers, database connections, warm standbys (etc.) regularly, it is recommended to create an `rs_init` resource file that can be used as a template. This allows you to run these common operations from the command line with `rs_init -r` instead of having to go through the interactive `rs_init` dialog every time. For more information on creating an `rs_init` resource file, see p.108.

## 2 RepServer basics

This section briefly describes some elementary aspects of RepServer:

- **RepServer** is Sybase's enterprise-level data replication product for log-based, publish-subscribe replication. RepServer provides best performance and functionality for ASE-to-ASE replication, but also supports heterogeneous replication (between different database types, like MS SQL Server, DB2 and Oracle).

- RepServer’s unit of replication is a transaction (not just an individual row modification): either the entire transaction is successfully applied to the replicate database, or it isn’t applied at all. Only committed transactions are replicated.

- By default, RepServer guarantees that replicated transactions will commit in the same order in the replicate database as they did in the primary database. When using HVAR or RTL in 15.5 (p.39), or MPR in 15.7 (p.106), this guarantee is relaxed as transactions can be merged or split up by RepServer.

- By default, RepServer replicates the effect of a DML statement (i.e. the changes to the data), not the DML statement itself. In 15.2, when enabling SQL state-
outside the database server (as for non-ASE databases).

8.1 ASE RepAgent configuration

The ASE RepAgent is a thread inside ASE. The ASE procedures in this section should be executed in the primary ASE database.

When RepServer has no connection for a particular database, but the ASE RepAgent tries to connect anyway, the ASE RepAgent will not start and ASE proc 9251 will appear in the ASE errorlog ("aborting due to an unrecoverable communications or Replication Server error"). To resolve this, either create a database connection in RepServer (see section 13; p.30), or disable the ASE RepAgent.

```
sp_config_rep_agent prod_db, 'send warm standby xacts', true
```

(sp procedure) Enables (1) or disables (0, =default) the ASE RepAgent (dynamic).

**(ASE 15.7)** `sp_config_rep_agent [database_name | NULL [, option, value [, ... more values... ]]]`

(ASE proc/ASE config) Enables (1) or disables (0, =default) the ASE RepAgent (dynamic). The number of 2KB memory pages for the ASE RepAgent (dynamic). Default= 4096 (=8MB); this may need to be increased when replicating many tables and columns.

```
sp_start_rep_agent database_name [, recovery_foreground] [, 'connect_database', 'connect_databases' [, 'repserver_name', 'repserver_username', 'repserver_password' ]]
```

(ASE procedure) Starts the RepAgent thread for `database_name`, where the secondary truncation point must be valid. For normal operations, only `database_name` is specified. To start the RepAgent in recovery mode, `recovery` must be specified; with `recovery_foreground`, progress messages are sent to the client. For the optional other options, see `sp_config_rep_agent`.

Once started, the RepAgent is automatically started after a future ASE restart.

```
sp_stop_rep_agent database_name [, 'nowait']
```

(ASE procedure) Stops the RepAgent for this database; `nowait` stops immediately. Once stopped, the RepAgent is not automatically started after an ASE restart.

```
sp_config_rep_agent [database_name | option, value, ...more values...]
```

(ASE procedure) Without parameters, lists the databases where a RepAgent is configured. With only `database_name`, displays all configuration settings for the RepAgent in that database. With `option`, displays the current setting; with also `value`, modifies the setting. Most changes take effect once the RepAgent is restarted. RepAgent config settings are stored in the `sysattributes` table in `database_name`. Examples: `sp_config_rep_agent prod_db, 'retry timeout', '20'` `sp_config_rep_agent prod_db, 'send warm standby xacts', true` option and value can be one of the following (note that is sufficient to specify a unique substring for option; also note that numeric values must be quoted):

- `enable`, `repserver_name`, `repserver_username`, `repserver_password` - configures the RepAgent for connecting to the specified RepServer with the specified username and password; also sets the secondary truncation point to valid in the database. `repserver_name` must be defined in the `interfaces` file used by this ASE server.

- `disable`, `preserve secondary truncpt` - removes the RepAgent configuration for this database. With `preserve secondary truncpt`, leaves the secondary truncation point unchanged; otherwise (=default), it is disabled.

- `rs servername`, `repserver_name` - the name of the RepServer to connect to.

- `rs username`, `repserver_username` - the RepServer username to be used for connecting to the RepServer.

- `rs password`, `repserver_password` - the password to be used for connecting to the RepServer.

- `connect datasource`, `dataserver_name` - when RepAgent is started in recovery mode to scan the log of a database dump (and/or log dumps) that were restored into a different database, specifies the name of the original dataserver.

Replication Server versions 15.1 - 15.7.1 21
This datatype conversion can be done on two levels:

- For an individual primary column (or parameter) in a repdef. For each column (or parameter), a datatype must be specified. By adding the map to datatype clause to the column declaration in the repdef (see above), all incoming values for that column will be translated to the specified datatype before being replicated further through the replication system.

  The official RepServer documentation calls this 'column-level translations'.

- For all data replicated to a replicate database. By creating datatype classes and datatype translations through the predefined SQL scripts mentioned below, a set of datatype translations is attached to a function string class. For a replicate database using that function string class, all data values replicated to that database are converted by the DSI according to those translations (only for those datatypes for which a translation has been defined).

  The official RepServer documentation calls this 'class-level translations'.

The two types of translations can be mixed. Column-level translations (map to in the repdef) are applied before class-level translations (performed by the DSI).

Setting up datatype classes/translations is simpler in 15.2 than in pre-15.2:

- 15.2 provides predefined connection profiles (>p.38). When specifying a connection profile name with create connection ... using profile, the required class-level translations are created automatically.

- In pre-15.2, to initialize datatype classes, datatype definitions and datatype translations for heterogeneous replication, certain predefined SQL scripts, located in the $SYBASE/$SYBASE_REP/scripts directory, must be executed in the RSSD by the DBA. See p.38 for more information.

11.3 RSSD tables - datatypes

Datatype classes are stored in the RSSD table rs_classes, with rs_classes.classstype = 'D'. For each datatype class, the RSSD table rs_datatype contains datatype definitions for that class.

The RSSD table rs_translation contains datatype translations for a particular function string class (the so-called 'class-level translations'). The RSSD table rs_columns contains the datatype conversions specified in the repdef with map to (the so-called 'column-level translations'), in columns declared_dtid (the declared datatype) and publ_dtid (the published datatype).

The RSSD rs_* procedures do not display any information about datatype translations. Instead, query the RSSD tables directly.

(Note: the exact relationship between datatype classes and function string classes remains unclear; in fact, there may not even be a direct relationship - the RepServer documentation just doesn't contain much information about this topic.)

In pre-15.2, 'empty' datatype classes, like rs_oracle_dt_class (in 15.5 also rs_iq_dt_class), are defined in the RSSD table rs_classes, with no corresponding rows in rs_datatype. Only the standard class rs_rs_dt_class is fully defined. The SQL scripts in $SYBASE/$SYBASE_REP/scripts insert rows into these RSSD tables, thus creating the datatype definitions and translations for these classes.

In 15.2, some (not all) additional datatype classes are defined by default. When a connection is created, additional definitions are added depending on the connection profile specified.

In principle, it should be possible to create your own datatype classes and corresponding translations. Because no RepServer commands exist for this purpose, it would require manually inserting rows into the RSSD tables rs_classes, rs_datatype and rs_translation. Unfortunately, exactly how to define such rows is mostly undocumented, but the SQL scripts mentioned above, and the RSSD table descriptions in the Replication Server Reference Manual (>p.14), may be a source of inspiration. Creating your own datatype classes and -translations is probably not officially supported by Sybase.

12 Connections to RepAgents

resuming distributor dataserver.database [ skip transaction ]

Resumes a suspended distributor (DIST thread) for a connection to a primary database. The distributor matches incoming primary transactions against repdefs and subscriptions, writing replicated transactions from the inbound queue into outbound queues. There is one DIST thread for each inbound queue.

skip transaction deletes the first transaction from the inbound queue; this may be needed when the distributor shuts down due to invalid incoming data (for example, when incoming primary data is incompatible with the corresponding repdef datatype). The skipped transaction is written into the RSSD exceptions log tables (>p.94), and
with rs_init as for any other database.

- An ASE RepAgent will be running in the primary master database, so its transaction log may need to be enlarged to ensure it does not fill up.

- The primary master database must be marked for replication with sp reptostandby 'master', 'all'.

- The ASE RepAgent in the primary master database must be configured with 'send warm standby xacts' = true (p.22)

- The master database can be replicated either as a warm standby or as a database replication definition and corresponding subscription.

- When using a database repdef for the ASE master database, the repdef properties replicate dll and replicate system procedures must both be enabled (or the DDL commands and sp_* calls above, respectively, will not be replicated).

- Materialization of the replicate master database should be done manually.

- Replication of the master database does not replicate the actual SuID/srID values for new logins and roles. Rather, it relies on identical contents for syslogins and syssrvroles on both sides so that the algorithms will pick the same values for new logins or roles.

- Since passwords for logins and roles are replicated as ciphertext, cross-platform synchronization is not supported (though it may actually work in some cases).

- Ensure that the sp_* procedures listed above are executed from within the master database if they are to be replicated.

**16.11 'send standby' option in repdefs**

'send standby' can be specified as an option in table repdefs (send standby all columns; send standby replication definition columns) and function repdefs (send standby all parameters; send standby replication definition parameters). This option applies only to a warm standby, and to database replication (MSA) when combined with table or function repdefs.

The absence or presence of send standby affects how primary data is replicated:

- **Replicated columns**: for a table repdef without a send standby option, or with only send standby (which is identical to send standby all columns), all columns of the primary table are replicated. With send standby replication definition columns, only the columns declared in the repdef are replicated.

- **Primary key columns**: for a table repdef with a send standby option (no matter which one), only the primary key columns of this repdef are used in the where-clause of replicate SQL update/delete statements. Without a send standby option, the union of all primary key columns for all repdefs for this table is used in the where-clause (Completeness note 1: without a repdef, all table columns are used in the where-clause of replicate SQL statements. Completeness note 2: when multiple repdefs exist for a table, only one repdef can have a send standby option enabled at a time).

- **Replicated parameters**: for a function repdef without a send standby option, or with only send standby (which is identical to send standby all parameters), all parameters of the primary stored procedure are replicated. With send standby replication definition parameters, only the parameters declared in the repdef are replicated.

For a warm standby only, the send standby options in the repdefs can be overridden by the configuration parameter send standby repdef cols for the logical connection (p.78). NB: the current send standby setting for a repdef is not displayed by any RSSD procedure (the columns Used by Standby of rs_helprep reflects if the repdef is used by a standby, but not the send standby clause).

**16.12 Replicating minimal columns**

'Minimal column replication' is a feature to improve performance for replicating update and delete statements on primary tables. Without minimal column replication, all columns for a modified row are replicated, including columns that have not changed. In contrast, minimal column replication only replicates those columns whose values have really been changed, in addition to the primary key columns. Especially for tables with many and/or long columns, minimal column replication can improve performance in the DSI (due to more efficient replicate SQL statements), over the network and in the stable queues (due to the replicated messages being shorter).

Minimal column replication is specified in a table repdef either as replicate all columns (disabled; all columns are replicated; =default) or replicate minimal columns
21.9 Subscribing to bitmaps

RepServer supports the special datatype `rs_address`, which can be used for subscribing to bitmaps (i.e. specific bits in a column or parameter). This can be useful when individual bits have a functionally relevant meaning, for example in status fields where a group of on/off indicators is packed together in one 4-byte integer.

**Warning:** before using bitmap subscriptions, please pay attention to the following. With bitmap subscriptions, a primary update will not be replicated if only `rs_address` columns are updated (unless a change in the bitmap would cause a row to migrate out of its current subscription or into a different subscription); this means that the bitmap values in replicate data rows could potentially get out of sync with the bitmap values in the primary rows. Therefore, you should verify that the actual update statements on your primary data are suitable for bitmap subscriptions.

To subscribe to specific bits, use the `&` operator and a bitmask value in the `where`-clause of `create/define subscription or create article`. This will perform a logical ‘AND’ on the actual value of the column/parameter and the bitmask when the result is non-zero, the row qualifies for the subscription.

In the example below, a bitmap indicates auxiliary employee skills. Let’s assume bit 0 indicates a 'First Aid' qualification, bit 1 indicates Dutch language ability and bit 2 indicates VMS experience (indeed, not a well-normalized relational data model…):

```sql
create replication definition EmployeeAuxSkills with primary at PROD.staff_db
(EmpNo int, EmpName varchar(50), Skills rs_address) primary key (EmpNo)
searchable columns (Skills)
create subscription FirstAid_sub for EmployeeAuxSkills
with replicate at ADMIN.office_mgmt_db
where Skills & 1 -- folks with a First Aid qualification…   (bit 0: 2^0 = binary 001)
and Skills & 2 -- … who also speak Dutch                   (bit 1: 2^1 = binary 010)
create subscription VMS_sub for EmployeeAuxSkills
with replicate at ADMIN.diehards_db
where Skills & 6 -- folks who speak Dutch or know VMS
-- (bits 1+2: 2^1 + 2^2 = binary 110)
```

Note the difference between the two `where` clauses: the first subscription requires both bits 0 and 1 to be set; the second subscription requires either bit 1 or 2, or both (this way, bitmap subscriptions can sometimes be used to simulate a logical 'OR' condition in the `where`-clause).

The bitmask can in principle also be specified as an 8-digit hexadecimal value: on Linux for example, `Skills & 0x00000006` is equivalent to `Skills & 6` above. However, this may not work cross-platform or on other platforms, since platform-specific byte ordering may produce unexpected results when using such hexadecimal values; with integer values, this problem does not apply.

rs_address represents a 4-byte (32-bit) datatype. Columns declared as `rs_address` in the repdef should normally be 4-byte integers in the primary table. `rs_address` columns cannot be part of the primary key clause in a repdef.

Although undocumented (and probably unsupported by Sybase), some other column datatypes -such as ASE's `smallint`, `tinyint` and `binary` (if not longer than 4 bytes)- can also be used for `rs_address` bitmaps; however, when using this, please note that you are fully responsible yourself for issues such as alignment and byte swapping!

21.10 Common subscription creation problems

When creating a subscription, the most common problems are:

- **Permission problems:** during materialization, the maintenance user does not have sufficient permissions for accessing tables in the replicate or primary database. Such permission problems invariably cause the DSI to be shut down; the RepServer errorlog will contain error messages with respect to this problem. When such permission problems occur, they can usually be solved by assigning the required permissions and resuming the DSI.

- **Disk partitions get full:** when a primary table is large, atomic materialization may cause the disk partitions to fill up (or at least, take a long time to complete). If this happens, one approach is to add as many disk partitions as needed, and be patient. However, it may be better to abort subscription creation (see below) and use a different method instead, like 'incremental atomic' materialization (p.64).

- **Replicate transaction log gets full:** when a primary table is large, atomic materialization may also cause the transaction log of the replicate database to fill up,
• 'derived class' - a function string class which has a parent class;
• 'base class' - a function string class which has no parent class.

A derived class inherits all function strings from its parent class; hence the name 'function string inheritance':
• When altering a function string in a parent class, this modification is immediately inherited by all derived classes for this parent class where this function string was not overridden (see next bullet).
• When creating a function string in a derived class, this overrides the inherited function string from the parent class (but does not affect the function string in the parent class itself).

When creating or modifying a function string, this takes effect immediately on the local RepServer; when there is an outgoing route, the modification takes effect in the destination RepServer once it has arrived in the destination RSSD.

Function string inheritance can have multiple levels: a parent class can also be a derived class at the same time. Parent classes can be established or changed with create function string class or alter function string class.

When creating a base class (i.e. a function string class without a parent class), all function strings must be explicitly created for the base class (with create function string); otherwise, any attempt to access a database with the function string class will cause errors in the DSI.

Notes:
• rs_sqlserver_function_class (the default class used for ASE connections) cannot be used as a parent class; use rs_default_function_class or another function class instead.
• Function strings cannot be created or modified for rs_default_function_class or rs_db2_function_class instead, create a derived class and create customized function strings in that class.
• Function string inheritance is dynamic (unlike the static class initialization for error classes with the RSSD procedure rs_init_erroractions: the latter is a one-time operation that is not maintained further).

23.4 Function string commands

{ create function string repdef_name.function_name(function_string_name) | create function string function_name } [for [(15.7.1) function class] function_class_name ]
[ with overwrite ]
[ scan 'input_whereclause_template' ]
[ output 
{ language 'output_sql_template' | rpc 'execute procedure_name [ @param ] = constant | ?variable|modifier? } [...] | writetext [ use primary log | with log | no log ] | none ]

Creates or modifies a function string for the specified function and function string class (a.k.a. a 'custom' function string), either for a table repdef or function repdef.

create function string…with overwrite is identical to alter function string.

(15.7.1)

create function string [owner.object_name.function_name;function_string_name] for database datasource.database …other clauses same as classic syntax above…

Creates or modifies a target-specific function string for a replicate table or stored procedure. Unlike the classic syntax above, these function strings are unrelated to a repdef or function string class, and can therefore be used in combination with table redefs, database redefs, or a warm standby. This can also be used together with automatic primary key detection (p.58). Target-specific function strings can be created for the same functions as for table repdefs, except for rs_select_[with_lock] (p.85); for stored procedures, the function name is identical to the procedure name. To display these function strings, use rs_helpobjfstring (p.85).

If the object is not owned by the maintenance user, the object owner should be specified or the function string won't work. No error is raised when the object doesn't exist.

Examples:

create function string Accounts_rdf.rs_delete for rs_sqlserver_function_class with overwrite output language
'delete Accounts where AccountNo = ?AccountNo!old? ;
insert LogTable values (getdate(), ?rs_origin_user!sys?,
"Deleted Account# ?AccountNo!old?")'
40 Configuration (.cfg) file

The RepServer configuration file (referred to as .cfg file in this book) is an ASCII file read by RepServer during startup to determine the server name, login and password for the ID server and RSSD server. Because the .cfg file contains passwords, it should be protected properly on file-system level.

By default, the .cfg file is located in $SYBASE/$SYBASE_REP/install and named repserver_name.cfg. The actual location of the .cfg file is passed to RepServer with the -C command line parameter in the RUN_servername file (p.116).

The .cfg file contains some rather self-explanatory keywords. It is not recommended to edit the .cfg file manually, unless you understand what you’re doing. When the .cfg file does not exist or contains invalid keywords, RepServer will not start. The .cfg file can also contain traceflags (p.110). When the .cfg file contains encrypted passwords, these passwords can only be modified through rs_init (p.108).

When a line starts with ‘#’, the entire line is considered a comment.

For RepServers with an embedded RSSD that have outgoing routes, the ERSSD RepAgent also has a file ending in .cfg, by default located in $SYBASE (p.95).

41 Replication system monitoring features

Over the years, RepServer has included different tools and features for monitoring the status of the various RepServers and dataservers in the replication system (not to be confused with the monitoring counters in section 47.2, which are about the internal workings of a single RepServer). A quick overview of the various abbreviations:

- RMP - Replication Manager (Plug-in for Sybase Central)
- RMS - Replication Monitoring Services
- RSM - Replication Server Manager (pre-15.0)
- RSP - RepServer Plug-in for Sybase Central (pre-15.0)

RMP and RMS are no longer part of the default RepServer installation, but can be downloaded from http://downloads.sybase.com (under ‘Replication Server’) and installed in the RepServer environment.

Most of the functionality of these tools has been moved into Sybase Control Center (SCC). Some aspects, such as reverse-engineering of RepServer constructs, is not supported by SCC, but still requires RMP (or Sybase PowerDesigner).

Sybase Control Center (SCC) is a web-based administration and monitoring tool for RepServer (it can also monitor ASE and IQ, supporting RS 15.0 and later. SCC must be downloaded from http://downloads.sybase.com (under ‘Sybase Control Center’) and installed separately (optionally, in the same RepServer environment or elsewhere; also optionally, on a dedicated host). Note that recent versions of RS, ASE and IQ install a directory $SYBASE/SCC-X_Y which in some cases do not contain the full SCC tool, but only the SCC agent that must be started (with $SYBASE/SCC-X_Y/bin/scc.sh) in order to monitor/manage a product through SCC. It may be best to always download and install the latest SCC version.

Once SCC is installed and started, go to https://scc-host:8283/scc (default login/password=sccadmin/blank; scc-host is the host on which SCC is installed. See the SCC manuals for further details.

42 Quiescing a replication system

A replication system is said to be ‘quiessed’ (or ‘quiestic’) when all RepServers have processed all incoming messages and delivered all outgoing messages, when there are no open transactions, and when no subscription (de)materialization is in progress. Quiescing a replication system or an individual RepServer is sometimes required, for example to modify the schema of a replicated table, or for troubleshooting.

To determine whether a RepServer is currently quiessed, run admin quiessce_check. Unless all log variance and incoming routes have been suspended, there is no guarantee that a quiessed state will be maintained.

To quiessce a replication system, perform these steps:

1. In all RepServers, run suspend log transfer from all.
2. In all RepServers with outgoing routes, run admin quiessce_force_rsi (p.97), forcing RepServer to send all messages in the queues for its routes. Wait briefly before proceeding with the next step, to allow the messages to be processed.
3. In all RepServers, run admin quiessce_check to determine whether the RepServer is quiessed. For those RepServers that are not quiessed, repeat step 2.
4. Finally, run steps 2 and 3 one last time.

Once a quiessed RepServer can return to its normal state, first resume log transfer for the RSSD (if the RSSD is replicated) before resuming log transfer for the other primary databases.
Index

#temporary tables (ASE) .............................................. 10
& (rs_address bitmask operator) ................................... 72
.cfg file (configuration file) ....................................... 127
32/64-bit .......................................................... 6, 98
alter applied function replication definition .............. 49
alter encryption key .................................................. 120
alter error class ....................................................... 93
alter function ........................................................ 91
alter function replication definition ......................... 49
alter function string .................................................. 84
alter function string class ......................................... 81
alter logical connection ............................................ 71, 74
alter partition ......................................................... 111
alter queue ............................................................ 44
alter request, funding replication def ....................... 49
alter role ................................................................ 96
alter schedule .......................................................... 143
alter subscription ....................................................... 70
alter use .................................................................. 118
alternate connection .................................................... 31, 31
always replicate (text/image data) .............................. 59
applied functions ....................................................... 47, 49
apply_truncate_table, syslogd ....................................... 71
articles, for publications ............................................ 62
ASE RepAgent .......................................................... 21
ASE Replicator ........................................................ 12
ASE Advanced Services Option .................................. 13, 39
assign action ............................................................ 92
async_parser (config) .................................................. 38
asyncfuncs, rs_ (RSSD table) ....................................... 84, 123
atomic materialization ............................................... 64
atomic materialization, incremental ......................... 65
audit_dest (config) ..................................................... 38
audit, for publications ............................................... 62
auditlog, location ....................................................... 137
ASE environment variables ....................................... 109
ASE RepAgent .......................................................... 21
ASE Replicator ........................................................ 12
ASE Advanced Services Option .................................. 13, 39
assign action ............................................................ 92
async_parser (config) .................................................. 38
asyncfuncs, rs_ (RSSD table) ....................................... 84, 123
atomic materialization ............................................... 64
atomic materialization, incremental ......................... 65
audit_dest (config) ..................................................... 38
audit, for publications ............................................... 62
auditlog, location ....................................................... 137
ASE environment variables ....................................... 109
automatic materialization ......................................... 65, 67
backlog, in queues ..................................................... 134
backup, of a RepServer .............................................. 146
base class ................................................................ 82
Basics, RepServer ...................................................... 59
batch (config) ............................................................ 33
batch_begin (config) .................................................... 34
batch_end, rs_ (function) ............................................. 87
batch_start, rs_ (function) ........................................... 87
begin, rs_ (function) ..................................................... 86
bitmap subscriptions .................................................. 72
block_size (config) ...................................................... 33
block_size (config) ...................................................... 33
block_size (config) ...................................................... 33
bulk_materialization ................................................... 65, 67
bulk_copy, dsi_ (config) .............................................. 36
bulk_threshold, dsi_ (config) ....................................... 36

The Complete Sybase Replication Server Quick Reference Guide

database generation number ........................................ 20, 135
database repdefs ......................................................... 51
  and autocorrection .................................................. 46
  function strings ...................................................... 53
  marking data for replication .................................... 17
  mixing with table/function repdefs .............................. 58
  switching primary and replicate ................................ 54
  vs. warm standby .................................................... 65

database replication .................................................. 71
  See database repdefs.............................................. 71
  databases, rs_ (RSSD table) .................................... 32, 79, 124
datarow_for_writetext, rs_ (function) .......................... 86
dataserver error class ............................................. 92
datareps, rs_ (RSSD table) .................................... 129

datatype, rs_ (RSSD table) ....................................... 86, 89

db2_error_class, rs_ (class) ..................................... 92
db2_function_class, rs_ (class) ................................ 81
db2_function_string .............................................. 81

db_packet_size (config) .......................................... 34
db2_function_string_class ....................................... 81
db2_function_string .............................................. 84
db2_function_string_replication_definition .................... 49
db2_function ......................................................... 91
diagnosis RepServer ............................................... 130
dictionary, rs_ (RSSD table) ..................................... 134
direct I/O (Q_DIRECT) ............................................. 102
direct routes ......................................................... 96, 108
direct_cache_read, dist_ (config) ................................. 101
DirectConnect .......................................................... 112

disk partitions ....................................................... 111

disk space requirements, estimate ................................ 112
disk affinity (config) .............................................. 33, 97, 111, 112
disk_space, admin .................................................... 55, 111
diskaffinity, rs_ (RSSD table) .................................... 112, 124
diskpartitions, rs_ (RSSD table) ................................ 112, 124

DIST thread (distributor) ........................................ 29, 36
dist_cmd, direct_replicate (config) ............................. 37
  dist_direct_cache_read (config) ............................... 101
  dist_sqt_max_cache_size (config) ............................. 35
  dist_stop_unsupp_cmd (config) ................................. 36, 76
  distributor (DIST thread) ..................................... 29, 36
  resume .............................................................. 29
  SOT optimization ................................................. 11
  suspend ....... 20, 30
  do not replicate (text/image data) ............................ 59

documentation about RepServer ................................ 14
drop function ......................................................... 91
drop function replication definition ................................ 49
drop function string .............................................. 84
drop function string class ........................................ 81
drop logical connection .......................................... 74
drop partition ......................................................... 111
drop publication ..................................................... 62
  drop replication definition .................................... 45
  drop route .......................................................... 95
  drop schedule ...................................................... 143
  drop subscription ................................................ 68, 70
  drop user .......................................................... 118
  drop_queue, sysadmin .......................................... 113
  dropdb, sysadmin ............................................... 32
  dropdropdb, sysadmin .......................................... 74
  dropps, sysadmin ................................................ 116
  DSI EXEC thread ................................................... 104
  DSI generation number ........................................... 135
  DSI thread .......................................................... 104
  multiple DSIs ..................................................... 106
  dsi_alt_writetext (config) ..................................... 36
  dsi_bulk_copy (config) .......................................... 36
  dsi_bulk_threshold (config) ................................... 36
  dsi_cdb_max_size (config) ...................................... 42
  dsi_charset_convert (config) ................................... 34
  dsi_check_thread_lock, rs_ (function) ......................... 87

  dsi_cmd_batch_size (config) ................................... 34
  dsi_cmd_fetch (config) .......................................... 34
  dsi_cmd_separator (config) ...................................... 34
  dsi_command_convert (config) .................................. 41
  dsi_commit_check_locks_intrvl (config) .................... 106
  dsi_commit_check_locks_log (config) ........................ 106

dematerialization .................................................. 66
derived class .......................................................... 82
destination RepServer (routes) ................................... 94
  destination_db, rs_ (variable) ................................ 89
  destination_ds, rs_ (variable) ................................ 89
  destination_lmdb, rs_ (variable) ............................... 89
  destination_lids, rs_ (variable) ............................... 89
  destination_ptype, rs_ (variable) .............................. 89
  destination_user, rs_ (variable) .............................. 89
  diagnostic RepServer ............................................ 130
The Complete Sybase Replication Server Quick Reference Guide

F
faudout_time, dsl_(config) .............34
fast_route_upgrade, sysadmin ..........99
dcfiablecap, rs_ (RSSD proc) ........112
filters, database repdef .................52
First Seg.Block (queues) ..............113
fioat (datatype) ..........................10, 27
forgotten, RS password ..................141
fungstrings, rs_ (RSSD table) .........84, 124
function repdefs ..........................47
mixing with database repdefs ............58
mixing with table repdefs ..............57
function string class inheritance .......81
function string classes .................80
function strings ..........................82
conflict resolution ......................90
custom ...................................82, 84
db repdef ..................................53
definition ................................79
input/output templates ..................88
modifiers (for variables) ..............88
restoring default .......................84
system variables ........................89
target-specific ...........................82, 84
various tricks ............................90
warm standby ............................78
functions
definition ................................79
function class-specific ..................86
function repdef-specific (user-def’d) 85
table repdef-specific ....................85
target-specific ................................82, 84
functions, rs_ (RSSD table) ............84, 84
G
gateway ....................................120
generation numbers ......................135
DSI ........................................133
primary database ........................145
get_charset, rs_ (function) ............87
get_generation, admin ...................136
get_lastcommit, rs_ (function) ..........87
get_lastcommit, rs_ (proc) .............33
get_sortorder, rs_ (function) ...........87
get_thread_seq, rs_ (function) ........86
get_thread_seq_noholdlock, rs_ (func) 87
getfunc, dbcc (ASE cmd) ...............20
grant .....................................149
ha_failover (config) .....................101
heartbeat ................................142
Help: replication doesn’t work! .......137
help_rep_agent_sp (ASE proc) ..........21
helpcfg, rs_ (RSSD proc) ..............100, 133
helpcfg_db, rs_ (RSSD proc) .........71, 78, 100
helpcfg_queue, rs_ (RSSD proc) .......115
helpcfg_route, rs_ (proc) ..............100
helpcheckrepdef, rs_ (RSSD proc) .....58
helpclass, rs_ (RSSD proc) ............81, 93
helpclassfstring, rs_ (RSSD proc) .....85
helpcounter, rs_ (RSSD proc) ..........135
helpdb, rs_ (RSSD proc) ...............33
helpdbrep, rs_ (RSSD proc) ..........59
helpdsub, rs_ (RSSD proc) .............71
helpererror, rs_ (RSSD proc) ..........93
helpexception, rs_ (RSSD proc) .......94
helpfstring, rs_ (RSSD proc) ..........84
helpfunc, rs_ (RSSD proc) ..............85
helpobjfstring, rs_ (RSSD proc) .......85
helppartition, rs_ (RSSD proc) .......111
helppub, rs_ (RSSD proc) ...............83
helppubsub, rs_ (RSSD proc) ..........61
helprep, rs_ (RSSD proc) ...............89
helprepdb, rs_ (RSSD proc) ..........33, 54
helpetable, rs_ (RSSD proc) ..........71
helprepversion, rs_ (RSSD proc) .....47
helproute, rs_ (RSSD proc) .............96
helpsub, rs_ (RSSD proc) ...............70
helpuser, rs_ (RSSD proc) ..............119
helpwerk, rs_ (RSSD proc) ..............124
heterogeneous RepAgents ................24
heterogeneous replication ...............12
Heterogenous Edition ....................14
hibernate_off, sysadmin ...............116
hibernate_on, sysadmin ................116
hibernation mode ......................116
HVAR
PK-FK constraints ........................43
PK-FK constraints in repdef ............45
retry failed transactions ...............40
ID server ....................................117
ID server user ...........................117
name of ID server ........................117
id_msc_confidentiality (config) ........104
id_msc_integrity (config) ...............104
id_msc_origin_check (config) ..........104
id_msg_seq_replay_detection (config) 104
id_msg_sequence_check (config) .......104
id_msc_auth (config) .....................104
id_security_mechanism (config) .......104
id_server (config) .......................101
id_userdef_login (config) .............104
identity columns ......................................61
identity columns, replicating .........61
identifiers, in primary database .......11
identifiers in primary database .........11
identity_columns, replicating ..........61
idames, rs_ (RSSD table) ...............124
ids, rs_ (RSSD table) .....................125
ignore loss ...................................130, 138
ignore_underscore_name(config) ......105
ignore_underscore_name(config) ......105
image/text data, mark for replication ....18
image/text/rowobject data .............59
IMDB, in ASE ..............................20
inbound queue .............................112
incremental atomic materialization ....65
indexes in replicate database ..........11
indirect routes ...........................96, 98
info, rs_ (table) ............................33
InfoPrimer .....................................41
Information Liquidity Model ............12
inheritance, function string classes ....81
init, rs_ (utility) .........................108
init_erroerrorations, rs_ (RSSD proc) ..93
init_scm_write_delay (config) .........101
init_scm_write_delay (config) .........101
initial RepServer username/password 116
initialize, new RepServer ..............108
initialize new RepServer ...............108
initialization of new RepServer .........108
insert, rs_ (function) .....................88
insert, rs_ (function) .....................85
The Complete Sybase Replication Server Quick Reference Guide

N
name, of ID server .......................... 117
name, of RepServer ........................ 26
native datatypes ................................ 27
net-change database (CDB) .... 39, 40, 42
network security .......................... 103
new, new_raw (variable modifier) .... 88
newgroups ....................................... 14
Next.Read (queues) ......................... 113
no materialization (mat.method) .... 64
non_blocking_commit, dsi_ (config) ... 37
non_blocking_commit, rs_ (function) .. 87
non_blocking_commit_flush, rs_ (func) 87
non-ASE RepAgent ........................... 24
non-atomic materialization ............ 65
NRM thread .................................... 35, 102
nrm_thread (config) ....................... 102
num_client_connections (config) .... 102
num_concurrent_subs (config) ......... 102
num_large_xact_threads, dsi_ (config) 105
num_msgqueues (config) .................. 101

O
objects, rs_ (RSSD table) ................. 59, 125
objfunctions, rs_ (RSSD table) ....... 84, 125
old, old_raw (variable modifier) ...... 88
opaque (datatype) ......................... 27
open transactions .......................... 27
OpenSwitch ................................... 12
OQID, layout .................................. 136
oqid, rs_ (RSSD table) ..................... 125
Oracle
ExpressConnect interface .................. 4
option for replication .................. 12
replicating to IQ ............................... 24
replication agent ......................... 24
warm standby .................................. 74
oracle_error_class, rs_ (class) ....... 32
oracle_function_class, rs_ (class) ..... 80
Origin queue ID ............................... See OQID
origin, rs_ (variable) ......................... 89
origin_begin_time, rs_ (variable) ...... 89
origin_commit_time, rs_ (variable) .... 89
origin_db, rs_ (variable) .................... 89
origin_ds, rs_ (variable) .................... 89
origin ldb, rs_ (variable) ................. 69
origin_lds, rs_ (variable) ................. 69
origin_qid, rs_ (variable) .................. 89
origin_user, rs_ (variable) ............... 89
origin_xact_id, rs_ (variable) ........... 89
origin_xact_name, rs_ (variable) ....... 90
oserver (config) .............................. 101
outbound queue ............................. 112
output templates (function strings) ... 88
parent class ................................... 81
partitioning rules, for parallel DSI ... 105
partitioning_rule, dsi_ (config) ....... 105
partitions, disk ............................. 111
password ........................................ 119
change, for maintenance user .......... 31
change, RepServer user ................. 118
for RepServer’s ‘sa’ default password... 116
reset ‘sa’ password ....................... 111
lost/forgotten .................................. 119
disable ........................................ 119
enable in rs_init ............................ 108
password expiration ...................... 108
password length (min, max) ............ 120
password policy settings ................. 120
password_encryption (config) ........... 119
password_encryption (config) ......... 119
passwords, rs_ (RSSD table) .......... 125
path ............................................. 108
See replication path
pdb_gen_id (non-ASE RepAgent) ....... 25
pdb_setreptable (non-ASE RepAgent) ... 25
pdb_setrepproc (non-ASE RepAgent) .... 25
pdb_setrepcol (non-ASE RepAgent) .... 25
pdb_truncate_log (non-ASE RepAg.) .... 25
perf_listrpts, rs_ (RSSD proc) .......... 135
performance tips ............................ 131
permissions ................................. 119
poster of RSSD tables ...................... 123
PowerDesigner ............................... 12
prepared statements
for a repdef and DSSIs ................. 44
for DSI ........................................ 36
crash RepAgent and DSI ................. 45
primary key
automatically detect ....................... 58
unique, why required ...................... 10
updating, issues ............................ 10
primary transaction log, full (ASE) .... 138
primary user
ERSSD ......................................... 121
RSDD/ERSSD ................................... 117
print rs_ticket (traceflag) ............... 111
profile, rs_ (RSSD table) ................. 39, 125
publications ................................. 62
publications, rs_ (RSSD table) ....... 64, 125
published datatype ...................... 28
publish-subscribe replication ........... 42
purge_all_open, sysadmin .............. 115
purge_first_open, sysadmin ............. 115
purge_normal_open, sysadmin .......... 115
purge_route_at_replicate, sysadmin ... 96
queue ID ........................................ See OQID
queue_dump_buffer_size (config) .... 102
queuemsg, rs_ (RSSD table) ............ 116, 125
queuemsgsxt, rs_ (RSSD table) .......... 116, 125
queues ........................................... 112
(de)materialization queues ............. 86
identifying ................................. 112
inbound, outbound .......................... 112
queue size and save_interval ........ 112
queue size and slow replication ....... 138
understanding queue growth ............ 113
queues, rs_ (RSSD table) ............... 116, 125
parallel DSI .................................... 104
parallel_dsi (config) ....................... 105
param, param_raw (variable modifier) 88
<table>
<thead>
<tr>
<th>Function/Procedure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rs_truncate</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_tvalues</td>
<td>(RSSD table)</td>
</tr>
<tr>
<td>rs_udb_function_class</td>
<td>(class)</td>
</tr>
<tr>
<td>rs_update</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_udb_error_class</td>
<td>(class)</td>
</tr>
<tr>
<td>rs_update_lastcommit</td>
<td>(proc)</td>
</tr>
<tr>
<td>rs_trunc_set</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_trunc_reset</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_translation</td>
<td>(RSSD table)</td>
</tr>
<tr>
<td>rs_triggers_reset</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_trunc_reset</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_ticket_v1</td>
<td>(proc)</td>
</tr>
<tr>
<td>save_interval</td>
<td>(config)</td>
</tr>
<tr>
<td>rs_subcmp</td>
<td>(utility)</td>
</tr>
<tr>
<td>rs_status</td>
<td>(table)</td>
</tr>
<tr>
<td>rs_statreport</td>
<td>(RSSD table)</td>
</tr>
<tr>
<td>rs_statdetail</td>
<td>(RSSD table)</td>
</tr>
<tr>
<td>rs_stat_populate</td>
<td>(RSSD proc)</td>
</tr>
<tr>
<td>rs_subscriptions</td>
<td>(RSSD table)</td>
</tr>
<tr>
<td>rs_session</td>
<td>(RSSD proc)</td>
</tr>
<tr>
<td>rs_textptr_init</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_stat Genreport</td>
<td>(RSSD proc)</td>
</tr>
<tr>
<td>rs_tvalues</td>
<td>(RSSD table)</td>
</tr>
<tr>
<td>rs_init_snapshot</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_ticket_report</td>
<td>(function)</td>
</tr>
<tr>
<td>rs_ticket_hist</td>
<td>(table)</td>
</tr>
<tr>
<td>rs_ticket_gen</td>
<td>(table)</td>
</tr>
<tr>
<td>rs_ticket_history</td>
<td>(table)</td>
</tr>
<tr>
<td>send_repserver_cmd</td>
<td>(proc)</td>
</tr>
<tr>
<td>set_log_name</td>
<td>(admin)</td>
</tr>
<tr>
<td>set_isolation_level</td>
<td>(function)</td>
</tr>
<tr>
<td>set_reptable</td>
<td>(config)</td>
</tr>
<tr>
<td>set_ciphertext</td>
<td>(function)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(proc)</td>
</tr>
<tr>
<td>show_route</td>
<td>(config)</td>
</tr>
<tr>
<td>show_connections</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_connection</td>
<td>(element)</td>
</tr>
<tr>
<td>show_server</td>
<td>(element)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(element)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_connections</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_reptable</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_repserver_cmd</td>
<td>(admin)</td>
</tr>
<tr>
<td>show_route_version</td>
<td>(admin)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>sysadmin upgrade</strong></td>
<td>98</td>
</tr>
<tr>
<td><strong>database</strong></td>
<td>98</td>
</tr>
<tr>
<td><strong>syssam (license manager)</strong></td>
<td>13</td>
</tr>
<tr>
<td><strong>SySAM (license manager)</strong></td>
<td>13</td>
</tr>
<tr>
<td><strong>sysmon, sp_ (ASE proc)</strong></td>
<td>23</td>
</tr>
<tr>
<td><strong>system procedures, replicated</strong></td>
<td>16</td>
</tr>
<tr>
<td><strong>system variables</strong></td>
<td>89</td>
</tr>
<tr>
<td><strong>system version</strong></td>
<td>98</td>
</tr>
<tr>
<td><strong>system_version.sysadmin</strong></td>
<td>98</td>
</tr>
<tr>
<td><strong>systext, rs_ (RSSD table)</strong></td>
<td>84, 94, 126</td>
</tr>
<tr>
<td><strong>table repdefs</strong></td>
<td>42</td>
</tr>
<tr>
<td><strong>mixing with database repdefs</strong></td>
<td>58</td>
</tr>
<tr>
<td><strong>mixing with function repdefs</strong></td>
<td>57</td>
</tr>
<tr>
<td><strong>using with warm standby/MSA</strong></td>
<td>58</td>
</tr>
<tr>
<td><strong>no repdef needed in 15.7</strong></td>
<td>58</td>
</tr>
<tr>
<td><strong>without subscriptions</strong></td>
<td>58</td>
</tr>
<tr>
<td><strong>targetobjs, rs_ (RSSD table)</strong></td>
<td>84, 126</td>
</tr>
<tr>
<td><strong>target-specific function string</strong></td>
<td>82, 86</td>
</tr>
<tr>
<td><strong>tbcfg, rs_ (RSSD table)</strong></td>
<td>39, 126</td>
</tr>
<tr>
<td><strong>templates, system variables</strong></td>
<td>89</td>
</tr>
<tr>
<td><strong>test_connection (non-ASE RepAgent)</strong></td>
<td>25</td>
</tr>
<tr>
<td><strong>text/image data, marking for replication</strong></td>
<td>18</td>
</tr>
<tr>
<td><strong>text/image data, marking for replication</strong></td>
<td>18</td>
</tr>
<tr>
<td><strong>text_status (variable modifier)</strong></td>
<td>88</td>
</tr>
<tr>
<td><strong>text_convert_multiplier, dsi_ (config)</strong></td>
<td>35</td>
</tr>
<tr>
<td><strong>text/image/rawobject data</strong></td>
<td>59</td>
</tr>
<tr>
<td><strong>transformation stored procedures</strong></td>
<td>49</td>
</tr>
<tr>
<td><strong>translation, rs_ (RSSD table)</strong></td>
<td>87</td>
</tr>
<tr>
<td><strong>triggers, and replication</strong></td>
<td>87</td>
</tr>
<tr>
<td><strong>triggers, rs_ (function)</strong></td>
<td>87</td>
</tr>
<tr>
<td><strong>triggers_reset, rs_ (function)</strong></td>
<td>87</td>
</tr>
<tr>
<td><strong>troubleshooting</strong></td>
<td>87</td>
</tr>
<tr>
<td><strong>connectivity problems</strong></td>
<td>137</td>
</tr>
<tr>
<td><strong>disk partitions full</strong></td>
<td>139</td>
</tr>
<tr>
<td><strong>general steps</strong></td>
<td>137</td>
</tr>
<tr>
<td><strong>transaction log full (ASE)</strong></td>
<td>138</td>
</tr>
<tr>
<td><strong>trigger, db_ (function)</strong></td>
<td>87</td>
</tr>
<tr>
<td><strong>trigger_table_partition</strong></td>
<td>87</td>
</tr>
<tr>
<td><strong>truncation point, secondary</strong></td>
<td>19</td>
</tr>
<tr>
<td><strong>tvalues, rs_ (RSSD table)</strong></td>
<td>126</td>
</tr>
<tr>
<td><strong>U</strong></td>
<td>80</td>
</tr>
<tr>
<td><strong>UDB vs DB2</strong></td>
<td>80</td>
</tr>
<tr>
<td><strong>udb_error_class, rs_ (class)</strong></td>
<td>92</td>
</tr>
<tr>
<td><strong>udb_function_class, rs_ (class)</strong></td>
<td>80</td>
</tr>
<tr>
<td><strong>Undocumented commands, disclaimer</strong></td>
<td>8</td>
</tr>
<tr>
<td><strong>Unicode</strong></td>
<td>37</td>
</tr>
<tr>
<td><strong>unicode_format (config)</strong></td>
<td>37</td>
</tr>
<tr>
<td><strong>unique primary key, why required</strong></td>
<td>10</td>
</tr>
<tr>
<td><strong>unlocking a locked user</strong></td>
<td>10</td>
</tr>
<tr>
<td><strong>unlocked</strong></td>
<td>120</td>
</tr>
<tr>
<td><strong>users, rs_ (RSSD table)</strong></td>
<td>126</td>
</tr>
<tr>
<td><strong>username/password, forgotten</strong></td>
<td>141</td>
</tr>
<tr>
<td><strong>username for RepServer, default</strong></td>
<td>116</td>
</tr>
<tr>
<td><strong>username/password, forgotten</strong></td>
<td>141</td>
</tr>
<tr>
<td><strong>username/password, forgotten</strong></td>
<td>141</td>
</tr>
<tr>
<td><strong>users, rs_ (RSSD table)</strong></td>
<td>126</td>
</tr>
<tr>
<td><strong>validate publication</strong></td>
<td>63</td>
</tr>
<tr>
<td><strong>validate subscription</strong></td>
<td>70</td>
</tr>
<tr>
<td><strong>varbinary_strip_trailing_zeros (config)</strong></td>
<td>37</td>
</tr>
<tr>
<td><strong>varchar_truncation (config)</strong></td>
<td>103</td>
</tr>
<tr>
<td><strong>verify_repserver_cmd, admin</strong></td>
<td>46</td>
</tr>
<tr>
<td><strong>version</strong></td>
<td>142</td>
</tr>
<tr>
<td><strong>version, admin</strong></td>
<td>142</td>
</tr>
<tr>
<td><strong>version, admin</strong></td>
<td>142</td>
</tr>
<tr>
<td><strong>version, rs_ (RSSD table)</strong></td>
<td>126</td>
</tr>
<tr>
<td><strong>W</strong></td>
<td>75, 143</td>
</tr>
<tr>
<td><strong>wait for create standby</strong></td>
<td>75, 143</td>
</tr>
<tr>
<td><strong>wait for failover</strong></td>
<td>75, 143</td>
</tr>
<tr>
<td><strong>wait for switch</strong></td>
<td>75, 143</td>
</tr>
<tr>
<td><strong>warm standby</strong></td>
<td>73</td>
</tr>
<tr>
<td><strong>warm standby</strong></td>
<td>73</td>
</tr>
<tr>
<td><strong>warm standby</strong></td>
<td>73</td>
</tr>
<tr>
<td><strong>warm standby</strong></td>
<td>73</td>
</tr>
<tr>
<td><strong>warm standby</strong></td>
<td>73</td>
</tr>
<tr>
<td><strong>warm standby</strong></td>
<td>73</td>
</tr>
<tr>
<td><strong>and autocorrection</strong></td>
<td>46</td>
</tr>
<tr>
<td><strong>DDL replication</strong></td>
<td>17</td>
</tr>
<tr>
<td><strong>function strings</strong></td>
<td>78</td>
</tr>
<tr>
<td><strong>getting status information</strong></td>
<td>79</td>
</tr>
<tr>
<td><strong>list of replicated DDL stmts/procs</strong></td>
<td>16</td>
</tr>
<tr>
<td><strong>marking data for replication</strong></td>
<td>17</td>
</tr>
<tr>
<td><strong>‘multiple warm standby’ (=MSA)</strong></td>
<td>51</td>
</tr>
<tr>
<td><strong>Oracle</strong></td>
<td>74</td>
</tr>
</tbody>
</table>
setup steps ........................................ 75
switch active, detailed steps ............. 77
switching applications to standby ...... 77
vs. database repdef ......................... 55
when .................................................. 78
where-clause
  display with rs_helppub ............... 63
  display with rs_helpsub ............... 70
  for a subscription ..................... 68
  for an article .............................. 63
whereclauses, rs_ (RSSD table) .... 64, 126
who, admin ........................................ 26
Windows
  registry key, startup parameters ... 110
  service, for RS .............................. 109
writetext, rs_ (function) .............. 86
writetext_log, rs_ (variable) ............ 90
writing strings to RS errorlog .... 27
ws_sqldml_replication (config) .... 78
xact_group_size, dsi_ (config) .... 34
zeroltm, rs_ (RSSD proc) .............. 20, 136