

The Complete Sybase IQ Quick Reference Guide

IQ versions 15.0, 15.1, 15.2, 15.3 & 15.4

These pages are taken from "*The Complete Sybase IQ Quick Reference Guide*", as a sample of the contents. The actual page size is 11 by 22 cm (4.3 by 8.6 inch).
The complete book can be ordered from <http://www.sypron.nl/iqqr> .

Rob Verschoor

Sypron Publications

The Complete Sybase IQ Quick Reference Guide

IQ versions 15.0, 15.1, 15.2, 15.3 & 15.4

by Rob Verschoor

ISBN 978-90-806117-0-2

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

Sypron is a registered trademark of Sypron B.V.

Sybase, Sybase IQ, Transact-SQL, Adaptive Server Enterprise and Replication Server are registered trademarks of Sybase, Inc. Other product or brand names may be (registered) trademarks of their respective owners.

Copyright © 2011 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.

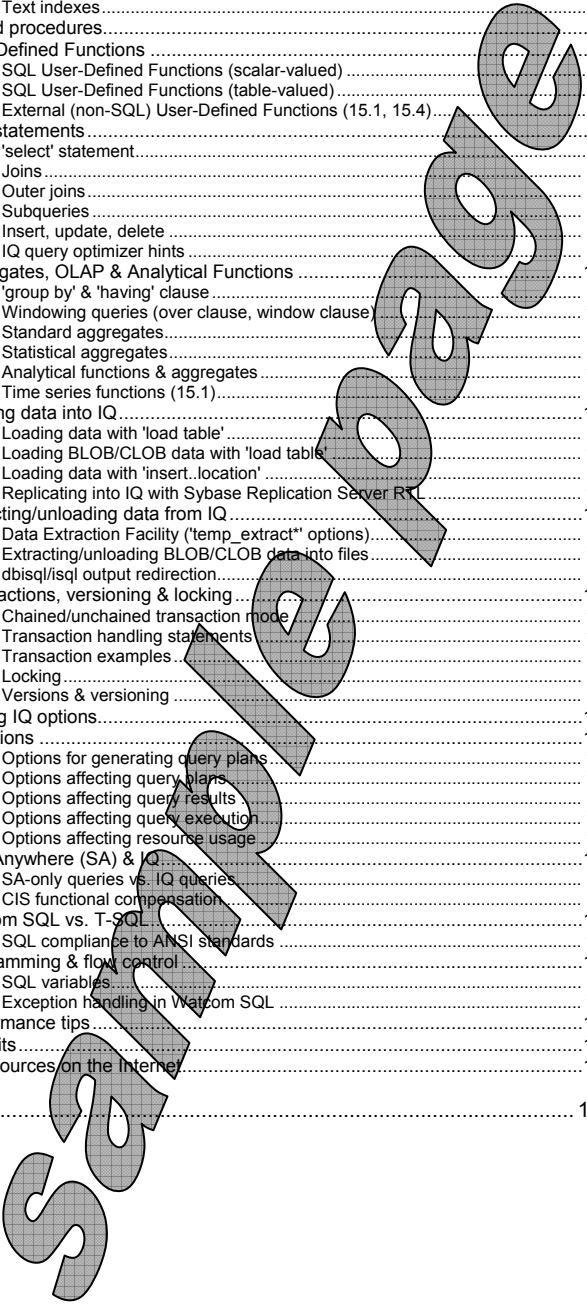
While this book has been prepared with care, neither the author, nor the publisher, nor Sybase Inc. (nor its subsidiaries), assume any responsibility for errors or omissions, nor do they accept any liabilities for damages resulting from the use of the information herein.

Contents

Introduction	6
About this Quick Reference Guide	6
IQ versions covered in this edition	6
How complete is 'Complete'? (topics not covered)	6
Sybase IQ for ASE users	7
How to use this Quick Reference Guide	7
Syntax conventions	7
Acronyms and Terminology	8
Errata	8
Electronic supplement	8
Undocumented statements: warning & disclaimer	8
Starting Points	9
1 Common IQ DBA tasks	9
2 Common IQ developer tasks	9
3 Sybase IQ: basic concepts	9
4 IQ quickstart for Sybase ASE users	10
5 Related Sybase products	11
DBA Topics	12
6 Getting info about server/database/connection	12
6.1 Server/database/connection properties	12
6.2 Server properties	13
6.3 Database properties	13
6.4 Connection properties	13
7 IQ editions	14
8 License Options	14
8.1 Managing license keys	15
9 Environment Variables	15
10 Starting IQ	16
10.1 Starting IQ in Utility Mode ('Utility Database')	17
10.2 Demo database ('iqdemo')	17
10.3 IQ error/message log files	18
10.4 Starting (and stopping) the IQ agent	19
10.5 Running IQ as a Windows Service	19
10.6 Starting multiple databases in an IQ server	20
11 Startup Switches	21
11.1 Commonly used startup switches	21
11.2 Security and permission-related switches	22
11.3 SA catalog-related switches	23
11.4 Message log-related switches	23
11.5 Multiplex-related switches	23
11.6 Resource-related switches	23
11.7 Troubleshooting switches	24
11.8 Other switches	24
12 Stopping IQ	25
13 Creating an IQ server	26
14 Creating an IQ database	26
14.1 'create database' statement	26
14.2 'iqinit' utility (15.2)	28
14.3 Collation (character set & sort order)	29
14.4 Renaming an IQ database	29
15 dbspaces & dbfiles	29
15.1 Block types reported by sp_iqdbspace/sp_iqfile	33
16 Remote Data Access (GIS/proxy tables)	34
16.1 Proxy tables mapping to external files	36
16.2 Getting information about remote servers, proxy tables	36
17 Users, Groups & Login Policies	36
17.1 Users	37
17.2 Login policies	38
17.3 Groups	39
18 Authorities & Permissions	40
19 IQ Multiplex	41
19.1 Creating a Multiplex	42
19.2 Adding/removing a node to a Multiplex	42
19.3 Sybase Central-generated admin scripts	42
19.4 Synchronizing secondary nodes	42
19.5 Multiplex statements & options	43
19.6 Multiplex startup switches	44

19.7	Distributed Query Processing 'PlexQ' (15.3)	44
20	IQ backups	46
20.1	Making backups	46
21	Restoring backups	47
21.1	Virtual backups	48
21.2	Getting information about backups	49
21.3	Backup/restore examples	49
21.4	Maintaining an IQ standby copy	50
22	Database Consistency Checks (DBCC)	50
22.1	Repairing corruptions	50
23	Monitoring	51
23.1	Monitoring user activity	51
23.2	Workload monitoring	52
23.3	Monitoring space usage	52
23.4	Performance-related monitoring	52
24	Miscellaneous DBA topics	53
24.1	Installation	53
24.2	DBA tools	54
24.3	IQ security features	54
24.4	Connection handle vs. Connection ID	55
24.5	Migration from IQ 12.x to IQ 15.x	55
24.6	Troubleshooting tips	55
24.7	Gathering information for Sybase TechSupport	56
Client Connectivity Topics		57
25	Client connections to IQ	57
25.1	Clients using Native IQ connections	57
25.2	Clients using OpenClient/TDS connections	57
25.3	Use Native IQ or OpenClient/TDS connectivity?	57
25.4	SQL behavior differences for IQ connection types	58
25.5	Determining the connection type	58
26	Native IQ connections	59
26.1	Connection parameters for native IQ connections	59
26.2	Switches or parameters in @filename.cfg or @variable	60
26.3	Connection parameters in \$SQLCONNECT	60
26.4	Connecting through the Shared Memory protocol	60
27	dbisql (Interactive SQL)	60
27.1	dbisql modes (GUI, -nogui, batch)	60
27.2	dbisql command-line options	61
27.3	dbisql commands	62
27.4	dbisql options	63
28	ODBC connections	64
29	OpenClient/TDS clients	65
29.1	Configuring OpenClient/TDS connectivity	65
30	Programming language APIs	65
Developer Topics		66
31	Comments	66
32	Identifiers	66
33	Strings, Quotes and Empty Strings	67
33.1	String comparisons	67
33.2	The empty string	67
33.3	Line-spanning strings	67
33.4	String-related options	67
34	Operators	68
35	Datatypes	69
35.1	User-Defined Datatypes (Domains)	71
35.2	Datatype conversion functions	71
36	Date & Time	72
36.1	Formatting date & time data	75
37	Mathematical Functions	76
38	Trigonometric functions	77
39	String functions	78
40	System functions	80
41	Tables	81
41.1	Temporary tables	83
41.2	Table partitioning	83
41.3	Constraints	85
41.4	Identity / autoincrement columns	86
41.5	Column Encryption	87
42	Views	87
43	Indexes	88
43.1	FP Indexes	88
43.2	Non-FP indexes	89

43.3	Indexing recommendations	90
43.4	Index advisor	92
43.5	Getting information about indexes	92
43.6	Join indexes	93
44	Full text search (15.2)	94
44.1	Full text search syntax	94
44.2	Text configuration objects	95
44.3	Text indexes	95
45	Stored procedures	96
46	User-Defined Functions	98
46.1	SQL User-Defined Functions (scalar-valued)	98
46.2	SQL User-Defined Functions (table-valued)	98
46.3	External (non-SQL) User-Defined Functions (15.1, 15.4)	99
47	DML statements	99
47.1	'select' statement	99
47.2	Joins	101
47.3	Outer joins	101
47.4	Subqueries	101
47.5	Insert, update, delete	102
47.6	IQ query optimizer hints	102
48	Aggregates, OLAP & Analytical Functions	102
48.1	'group by' & 'having' clause	103
48.2	Windowing queries (over clause, window clause)	103
48.3	Standard aggregates	104
48.4	Statistical aggregates	105
48.5	Analytical functions & aggregates	106
48.6	Time series functions (15.1)	107
49	Loading data into IQ	107
49.1	Loading data with 'load table'	107
49.2	Loading BLOB/CLOB data with 'load table'	110
49.3	Loading data with 'insert..location'	111
49.4	Replicating into IQ with Sybase Replication Server RPL	112
50	Extracting/unloading data from IQ	112
50.1	Data Extraction Facility ('temp_extract*' options)	112
50.2	Extracting/unloading BLOB/CLOB data into files	113
50.3	dbisql/isql output redirection	113
51	Transactions, versioning & locking	113
51.1	Chained/unchained transaction mode	114
51.2	Transaction handling statements	114
51.3	Transaction examples	115
51.4	Locking	116
51.5	Versions & versioning	117
52	Setting IQ options	117
53	IQ options	119
53.1	Options for generating query plans	119
53.2	Options affecting query plans	120
53.3	Options affecting query results	121
53.4	Options affecting query execution	122
53.5	Options affecting resource usage	123
54	SQL Anywhere (SA) & IQ	124
54.1	SA-only queries vs. IQ queries	124
54.2	CIS functional compensation	125
55	Watcom SQL vs. T-SQL	125
55.1	SQL compliance to ANSI standards	126
56	Programming & flow control	126
56.1	SQL variables	129
56.2	Exception handling in Watcom SQL	129
57	Performance tips	130
58	IQ limits	131
59	IQ resources on the Internet	131
	Index	132



purposes. Higher settings consume more memory, so do not set unnecessarily high. When higher than 1001, IQ will not start. **-gm 1** starts IQ in single-user mode; this is often combined with **-gd dba** to avoid interference by non-**dba** users.

To restrict access to 1 connection after starting IQ normally, use **sa_server_option 'disable_connections', 'on'**; when **'off'**, normal access is restored (this sets the server property **ConnsDisabled**).

-iqmt *nr-of-threads*

Number of threads to create for the IQ server. Default= approx. 60*nr-of-CPU's + setting of **-gm**; max. value is 4096 (64-bit IQ) or 2048 (32-bit). The actual #threads is logged during IQ startup in the **.stderr** file (>p.18).

-gt *nr-of-CPU's*

-gtc *nr-of-CPU's*

-gt defines the maximum number of physical CPU's that may be used by IQ, and **-gtc** the maximum number of logical CPU's (i.e. when using hyperthreading).

-iqnumbercpus *nr-of-CPU's*

Defines the number of CPU's (=cores) available to IQ (1-512), overriding the physical CPU's (=cores) IQ detects itself. This option is only to help the optimizer to run parallel queries better, or when physical CPU availability to IQ has been restricted.

-p *size-in-bytes*

Maximum size of network packets for client connections to IQ, in bytes. Default=1460; min=500, max=16000. Native IQ clients can independently set the packet size with **CommBufferSize (cbsize)** connection parameter (max=16000). OpenClient/TDS connections must use a packet size below the server maximum specified with **-p**.

-pc

-pt *minimum-size*

For native IQ connections, **-pc** will compress client-server network packets(except when the client runs on the same host as the IQ server); **-pt** specifies the *minimum-size* (in bytes) of a packet in order to be compressed. Default *minimum-size* is 120.

11.2 Security and permission-related switches

-gd { *dba* | *all* | *none* }

Defines who can explicitly start/stop a database (with **start database** or **restore backup**). The default **.cfg** file has **-gd all** (=any user); other values are **dba** (=dba authority) and **none** (=nobody). The setting is in server property **StartDBPermission**.

-gk { *dba* | *all* | *none* }

Defines who can stop the IQ server with **stop engine** or the **dbstop** utility. Default=**dba** (=dba authority); other values are **all** (=any user); **none** (=nobody; IQ can only be stopped with **stop iq**). When IQ is started with **-gk all**, the IQ documentation speaks of a "personal server".

-gl { *dba* | *all* | *none* }

Defines who can run **load table**. The default **.cfg** file has **-gl all** (=any user); other values are **dba** (=dba authority) and **none** (=nobody). Note that **alter** permission on the actual table is also required to run **load table**.

-gu { *dba* | *all* | *none* | *utility_db* }

Defines who can run **create/drop database**. Default=**dba** (=dba authority); other values are **all** (=any user); **none** (=nobody) or **utility_db** (=only the **dba** user, when connecting to the **utility_db** database with the password in **util_db.ini**; >p.17).

-su *password*

Specifies the **dba** user's password for connecting to the utility database (>p.17).

-ek '*encryption-key*'

Encryption key for an IQ database that was created with strong encryption (>p.27). Must be specified on the **start iq** command line following the **.db** filename; cannot be specified in the **.cfg** file.

-ep

On Windows only, prompts for the encryption key for an IQ database that was created with strong encryption (>p.27).

-ec [*none*],[*simple*],[*tls(TLS-specification)*]

For native IQ connections only, specifying **simple** or **tls(...)** enables encrypted client-

the amount of internal IQ-to-SA communication (also, >p.125).

When specifying a number, displays only the connection with that connection handle.

sp_iqcontext [*connection-handle*]

Without parameters, displays information about currently executing queries. The currently executing SQL is displayed in column **CmdLine**.

When specifying a number, displays only the connection with that connection handle.

sp_iqclient_lookup ['*client-IP-address*' [, *client-IP-port* [, '*user-name*']]]

Without parameters, for all currently active connections, displays the client-side IP address and port number of the client's connection (the only item not shown by **sp_iqconnection**), and the user name. When specified, the parameters act as filters; specifying **NULL** matches all connections.

23.2 Workload monitoring

Workload by client applications can be monitored in various ways:

- 'request logging' using startup switches **-zr sql** and **-zo** (>p.19). This collects the actual SQL statements sent to the IQ server by client applications.
- **sp_iqcontext** (>p.52) displays currently executing SQL.
- **sp_iqworkmon** (below) collects information about usage of tables, columns and indexes in queries.

sp_iqworkmon ['*action*'] [, '*monitor-mode*']

Collects information about usage of IQ tables, columns and indexes in **select/update/delete** queries. SA tables are ignored by **sp_iqworkmon**.

action can be **start**, **stop**, **status** (=default, shows current status and #distinct tables/columns/indexes items counted), or **reset** (resets collected information to 0). *monitor-mode* indicates what is monitored and can be **table**, **column**, **index** (for non-FP indexes) or **all** (= all three). The collected information (number of accesses and the time of the last access) can be viewed with the following stored procedures:

- **sp_iqtableuse**: reports access count for IQ tables.
- **sp_iqcolumnuse**: reports access count for columns in IQ tables.
- **sp_iqindexuse**: reports access count for non-FP indexes.

Stored procedures **sp_iqunusedtable**, **sp_iqunusedcolumn** & **sp_iqunusedindex** report tables, columns and indexes that were not accessed by queries while **sp_iqworkmon** was active (since the last **reset** action).

The collected information is kept only in memory and is lost after an IQ reboot.

23.3 Monitoring space usage

For information about the amount of space being used, use these procedures:

- **sp_iqspaceused** (>p.28): reports total size and amount used for the database
- **sp_iqdbspace** (>p.32): reports total size and amount used for each dbspace
- **sp_iqfile** (>p.32): reports total size and amount used for each dbfile

For more information about the tables, indexes, partitions etc. that are consuming space, see the list of stored procedures on p.32.

23.4 Performance-related monitoring

Various aspects of IQ performance and resource usage can be monitored with **sp_iqsysmon**, **iq utilities** and **sp_statistics**; **sp_iqstatus** also provides some information. In addition, Sybase Central (>p.54) has a graphical performance monitor (right-click the IQ icon for the IQ server; then, in the right-hand window, select the 'Performance Monitor' tab).

sp_iqsysmon [...*options*...]

Based on **iq utilities** (see below), reports details on IQ's buffer management, CPU utilization, and other aspects related to resource usage and performance.

sp_iqsysmon can be used in various ways:

- **sp_iqsysmon 'hh:mm:ss' [, '*sections*']**
Displays a report over the specified time interval; optionally, *sections* limits the output (see below).
- **sp_iqsysmon start_monitor**
... run some workload...
sp_iqsysmon stop_monitor [, '*sections*']
Displays a report over the interval between **start_monitor** and **stop_monitor**; optionally, *sections* limits the output (see below).
- **sp_iqsysmon start_monitor, filemode** [, '*monitor-options*']
... run some workload...
sp_iqsysmon stop_monitor

Example: **create table MyTable (a varchar(10) null);**
insert MyTable values ('');
 -- returns 1 row when **non_ansi_null_varchar=on**; 0 rows when **off**.
select * from MyTable where a is null;

string_truncation (option, on/off)

When **off** (=default), no error is raised when a character string is truncated by an **insert** or **update** command of a **var[char]** column, or assignment to a **var[char]** variable. When **on** (=default, =ANSI SQL-compliant), an error is raised only when non-space characters are truncated; also, the specific **insert/update** causing the truncation is rolled back (but the transaction continues). NB: in pre-15, the default was **off**.
set string_truncation on/off (T-SQL syntax) is equivalent to **set temporary option string_truncation=on/off**.

escape_character (option, on/off)

When **on** (=default for native IQ connections), a backslash character (\) is interpreted as an escape character rather than as a normal character. This escape is required for specifying non-printable characters like '\x0a' or '\n'. Also, when **on**, double backslashes must be used for Windows pathnames (i.e. **C:\\Windows\\System32**). When **off** (=default for OpenClient/TDS connections), the backslash is a regular character with no special meaning.

tds_empty_string_is_null (option, on/off)

For TDS connections only (it is ignored by native IQ connections), affects the client-side representation of an empty string value. When **off** (=default) an empty string is converted to a single space; when **on**, to a **NULL** string. How these values are eventually formatted depends on the particular client application. Note that the conversion is performed by the TDS driver, after all server-side SQL processing has been performed (i.e. the SQL results are not affected by **tds_empty_string_is_null**).

ansi_blanks (option)

In embedded SQL, affects string truncation when reading data into host variables.

34 Operators

Numeric/mathematical operators

For all numeric datatypes, the operators **+**, **-**, *****, **/** and **%** (modulo) are supported. For additional mathematical functions, such as **abs()** and **power()**, >p.76.

ASE-IQ compatibility: In IQ queries, a numeric expression cannot contain **NULL** (e.g. **3+NULL**). In ASE, and in SA-only queries, this evaluates to **NULL**.

String operators

Operators **||** and **+** both concatenate two strings (for **[var]binary** as well as character strings). **||** converts all operands to character strings, while **+** does no conversions.

|| and **+** cannot handle **long varchar** data; these must be converted to **[var]char** first.

Examples: **select 'Right now is: ' || now()**

select ' Right now is: ' + cast(now() as varchar)

For string functions such as **like substring()**, >p.78.

Date & time operators

For date & time-related datatypes, the **+** and **-** can be used in different ways:

- For **date** or **timestamp** datatypes, when used with a numeric expression, the **+** and **-** operators add and subtract, respectively, a number of days.
- For **date** or **timestamp** datatypes, when used with another **date** or **timestamp**, the **-** operator calculates their difference, as an integer number of days.
- When used with a **date** and a **time** datatype, the **+** operator combines both into a **timestamp** datatype.

Examples: **select '36 hours from now' = now() + 1.5;**

select '4 weeks ago' = now() - 28;

select 'days as an SAP company' = now()-cast('May 12 2010' as date);

For date & time functions (like **now()**), >p.72.

Logical operators

Logical operators are used in logical expressions (such as in predicates), which evaluate to true or false (except when involving **NULL** with **ansinull=on**, when the result can also be unknown):

- **=, <>, !=, >, >=, <, <=, !>** (not greater than), **!<** (not less than) - comparisons
- **and, or** - combines two logical expressions: **if (@a > @b) and (@x != @y)**
- **not** - negates a logical expression: **if not (@a > @b)**
- **[not] like** - a pattern matching operator for strings; see below.
- **contains()** - used with **WD** (word) and text indexes, >p.91, p.94.
- **[not] in** - matches a list of values: **where a in (3,4,5)** - this is equivalent to **where a=3 or a=4 or a=5**

cache) than this number (in MB) are not executed, but raise an error. Default = 0 = no limit. **sp_iqshowpsex** shows the current setting.

In 15.3, for a distributed query, this limit applies to the sum of **iq_shared_temp** space and **iq_system_temp** space used by all MPX nodes participating in the query.

max_temp_space_per_connection (option)

When set to a number > 0, any statement or load operation causing the total amount of temp store space (not temp cache) used by the current connection to exceed this number (in MB), will be aborted during execution. Default = 0 = no limit. **sp_iqshowpsex** shows the current setting. This is a run-time limit, as opposed to **query_temp_space_limit** which applies prior to execution.

In 15.3, for a distributed query, this limit applies to the sum of **iq_shared_temp** space and **iq_system_temp** space used by all MPX nodes participating in the query.

sp_iqshowpsexiq [*connection-handle*]

For all connections, or only for the specified connection, displays information about option settings for task priority and resource limits (like **iqgovern_priority** or **max_query_time**).

53.5 Options affecting resource usage

The options below affect the amount of resources available when executing a query.

-iqgovern max-nr-statements (startup switch)

Defines the maximum number of statements that can be executed concurrently (across all users) in the IQ server. Default=2*(nr of CPUs)+10. Setting this value too high or too low may affect overall performance.

iqgovern_priority (option)

Defines the execution priority for a query waiting to be executed. Values are 1 (high priority), 2 (medium,=default), and 3 (low); the query is placed into a queue with corresponding priority; all queries in higher-priority queues are executed before moving to a lower-priority queue. The current setting for a connection is displayed by **sp_iqshowpsex** as HIGH/MEDIUM/LOW (for 1/2/3).

iqgovern_priority cannot be set lower than option **iqgovern_max_priority**, whose default is 2 (otherwise an error message is raised, mentioning only 'invalid setting').

iqgovern_priority_time (option)

Defines the number of seconds a priority 1 query waits in the queue before starting execution. When reaching the time limit, the query starts even if this exceeds the max. number of concurrent statements defined by the **-iqgovern** switch. Default = 0 = no waiting; maximum is 1000000 (seconds, = 11 days).

max_query_parallelism (option)

Defines the maximum level of parallelism allowed for query operators. Default=64; maximum=512. When 1, parallelism is disabled (not recommended). It can sometimes be beneficial to use a value higher than the actual number of cores available to IQ.

max_iq_threads_per_connection (option)

Defines the maximum number of processing threads available to a connection for use in IQ operations. Values are 3-10000, default=144.

max_iq_threads_per_team (option)

Defines the maximum number of processing threads for a particular operation in a query. Values are 1-10000, default=144. The total #threads for a parallel processed query must remain below option **max_iq_threads_per_connection**.

max_statement_count (option)

Defines the max. number of prepared statements that can be used per connection. Default = 100; when 0, no limit applies. **sp_iqshowpsex** shows the current setting.

max_cursor_count (option)

Defines the maximum number of cursors that can be used per connection. Default = 50; when 0, no limit applies. **sp_iqshowpsex** shows the current setting.

hash_pinnable_cache_percent (option)

Defines the maximum percentage of the currently available temp cache that a hash operation can allocate (a run-time limit). Values are 0-100, default=20. This is an advanced setting; higher values may impact other user's queries.

Index

Legend

- Index entries followed by **' , sp_ ' , ' sp_iq ' or ' sa '** are stored procedures where the **sp_ , sp_iq or sa_** prefix has been chopped off to make a better index reference; for example, **connection , sp_iq** refers to **sp_iqconnection**.
- To maximize the usefulness of the index, index entries are often followed by **(option) , (dbisql command) , (aggregate) ,** etc. as a quick hint about the meaning of the entry.

0-9

1099, default IQ agent port..... 19
 2638
 broadcast port number 21
 default IQ port number 17, 21

. @

#temporary tables..... 83

 .backup.syb file..... 49
 .cfg file 60
 .db file..... 27
 .iq file..... 27
 .iqmsg file 18, 27
 rename..... 18
 .iqtmp file..... 27
 .isqlPreferences11 file (dbisql)..... 63
 .lmp file..... 15
 .log file..... 27
 .NET..... 65
 .srvlog file..... 18
 .stderr file..... 18

 @@error 130
 @@identity..... 86
 @@isolation..... 115
 @@procid..... 97
 @@servername 13, 21
 @@spid 55
 @@tranchained 114
 @@trancount..... 115
 @@transtate..... 114
 @file-or-variable (for conn. params)..... 60

A

abs()..... 76
 acos()..... 77
 acronyms used in this book 8
 addgroup, sp_..... 39
 addlogin, sp_..... 37
 addlogin, sp_iq..... 37
 addtype, sp_..... 71
 adduser, sp_..... 39
 ad-hoc join..... 93
 ADO.Net..... 6
 advice for indexing..... 92
 aes_decrypt()..... 87
 aes_encrypt()..... 87
 agent
 IQ agent..... 19
 SCC agent..... 54
 aggregate functions
 system-defined aggregates..... 102
 user-defined aggregates..... 99
 aggregation_preference (option) 120
 all (opposite of distinct) (operator)..... 100

all (subquery operator)..... 101
 allow_nulls_by_default (option)..... 58
 allow_read_client_file (option)..... 110
 alter database upgrade..... 28
 alter dbspace 30, 31
 alter function 98
 alter logical server..... 45
 alter login policy..... 38, 43
 logical server..... 45
 alter ls policy..... 45
 alter multiplex rename..... 43
 alter multiplex server..... 43
 alter procedure..... 96
 alter server..... 34
 connection close..... 35
 alter table.....
 add..... 82
 add constraint..... 85
 alter default..... 82
 alter null/not null..... 82
 drop..... 82
 drop check..... 85
 drop constraint..... 85
 drop default..... 82
 drop partition..... 84
 merge partition..... 84
 move..... 32
 partition..... 83
 rename..... 82, 84, 85
 split partition..... 84
 unpartition..... 84
 alter text configuration..... 95
 alter text index..... 95
 alter user..... 37, 38
 alter view..... 88
 and (logical operator)..... 68
 ANSI join syntax..... 101
 ANSI standard, compliance..... 126
 ansi_blanks (option)..... 58
 ansi_standard_packages, sa_..... 126
 ansi_substring (option)..... 79
 ansinull (option)..... 58, 122
 any (subquery operator)..... 101
 API, for programming languages..... 65
 append_load (option)..... 110
 argn()..... 81
 ascii()..... 78
 ase_function_behavior (option)..... 72
 asin()..... 77
 atan()..... 77
 atan2()..... 77
 atn2()..... 77
 audit_string, sa_..... 55
 auditing (option)..... 54
 authorities..... 40
 auto_commit (dbisql option)..... 63
 autoincrement column..... 86
 automatic startup
 IQ agent..... 19
 IQ as Windows service 20
 IQ on Unix/Linux 16

avg() (aggregate).....	104	sp_[iq]password	37
		changegroup, sp_	39
		char().....	78
B		char_length().....	80
backslash (escape_character option) ..	68	char_length64().....	80
backup database	46	char_terms, sa_	95
full/incremental backups	46	character set	
selective backup	46	for database	29
backup.syb file.....	49	for IQ server	29
backupdetails,sp_iq	49	list available	29
backupsummary, sp_iq.....	49	used by client	29
base tables	81	charindex().....	78
batch mode (dbisql).....	61	check (constraint).....	85
bcp (OpenClient utility)	65	checkdb, sp_iq	
begin parallel iq...end parallel iq	126	check consistency/corruptions.....	50
begin transaction	114	repair corruptions	51
begin...end	126	resetlocks (MPX)	44
between (logical operator)	69	checkoptions, sp_iq	119
bfile()	113	checkpoint.....	115
big endian	109, 113	checkpoint_execute, sa_	48
big SQL queries (SQL text size)	23	checkpoint_time (option).....	115
biginttohex().....	72	CIS.....	125
bit operators.....	69	CIS functional compensation.....	125
bit_length().....	80	cis_option (option).....	34
bit_vector_pinnable_cache_%(option).....	124	-cl (catalog cache).....	23
BLOB data		client_lookup, sp_iq.....	52
blob (long binary datatype)	69	CLOB data	
extracting/unloading	113	clob (long varchar datatype).....	69
loading.....	110	extracting/unloading	113
block types, in sp_iqdbspace/sp_iqfile ..	33	loading	110
blocking locks		cloning an IQ database	26
blocking (option).....	116	CmdSeq (command sequence).....	57
display, with sa_conn_info.....	116	CMR index	90, 91
waiting for, with lock table.....	116	coalesce()	127
break (from loop)	128	col_length()	81
broadcast (port 2638)	21, 24, 59	col_name()	81
buffer thrashing.....	122	collation	
bye (dbisql command)	62	for database	29
byte_length().....	80	list available	29
byte_length64().....	80	column encryption	87
byte_substr().....	79	column name list for load table.....	110
byte_substr64().....	79	column, sp_iq.....	82
		columnuse, sp_iq	52
C		command line, startup	17
-c (catalog cache size).....	23	command prompt mode (dbisql).....	61
-c (native IQ connection parameters).....	59	command sequence (CmdSeq).....	57
cache size		comment	
catalog cache (-c, -cl, -ch)	23	in SQL, comment delimiters	66
main cache (-iqmc).....	21	on objects.....	66
temp cache (-iqtc).....	21	commit transaction/work.....	114
cache_partitions (option).....	23	commit_on_exit (dbisql option).....	63
call	96	Component Integration Services(CIS) ..	34
cardinality_analysis, sp_iq	91	composite index (HG, CMP).....	90
cartesian product	122	conn_info, sa_	116
case		conn_properties, sa_	14
case expression.....	127	connecting to IQ	57
case statement	127	connection handle vs. connection ID ..	55
case-sensitivity	27, 66	connection parameters.....	59
cast()	72	connection properties	13
catalog cache size (-c, -cl, -ch)	23	connection type, determine	58
catalog store (system)	30	connection, sp_iq	51
catalog store page size	27	connection_property()	13
ceil().....	76	constraint, sp_iq.....	85
ceiling().....	76	constraints	85
-ch (catalog cache).....	23	ignore, by load table	109
chained (option).....	58, 114	contains()	
chained transaction mode	114	with text index	91, 94
change password		with WD (word) index	91
alter user	37	context, sp_iq.....	52
		continue (in loop).....	127, 128
		conversion_error (option)	72

conversion_mode (option)	72	date & time	72
convert()	71	datatypes	69
for date/time formatting	75	formatting	75
cooperative_commits (option)	115	functions & options	72
coordinator node, (MPX)	41	operators	68
copyloginpolicy, sp_iq	38	DATE index	90, 91
core_options67 (option)	89	date()	72
corr() (aggregate)	105	date_first_day_of_week (option)	73
correlation, aggregate	105	date_format (option)	58
corruptions	50	date_order (option)	58, 76
cos()	77	dateadd()	74
cot()	77	dateceiling()	74
count() (aggregate)	104	datediff()	74
count(*) (aggregate)	104	datefirst, set	74
covar_pop() (aggregate)	105	datefloor()	74
covar_samp() (aggregate)	105	dateformat, set	76
covariance, aggregate	105	datename()	74
create database	26	datepart()	74
create dbspace	30	dateround()	74
create existing table	35	datetime()	72
create externlogin	35	day()	73
create function	98	dayname()	73
create global temporary table	83	days()	73
create index	90	db_backupheader (utility)	49
create join index	93	db_id()	20, 81
create local temporary table	83	db_name()	20, 81
create logical server	45	db_properties, sa	13
create login policy	38	DBA tools	54
create message	130	dbcc	See sp_iqcheckdb
create multiplex server	43	dbcc_log_progress (option)	51
create procedure	96	dbcc_pinnable_cache_percent(option)	51
remote procedure	36	dbfiles	29
create server	34	dbisql (utility)	60
mapped to filesystem	36	batch mode	61
create table	95	command line options	61
partitioned tables	83	commands (dbisql-specific)	62
proxy tables	35	GUI mode	61
create text configuration	95	nogui mode	61
create text index	95	not all results displayed	62
create user	37	options (dbisql-specific)	63
create variable	129	output redirection	113
create view	87	query plan viewer	120
create wd index	90	dbisqlc (utility)	60
cross join	101	dblog (utility)	48
cube, in group by clause	103	dbo (group)	39
cume_dist() (OLAP)	106	dbping (utility)	64
current database	20	dbspace, sp_iq	32
current date	72	dbspaceinfo, sp_iq	32
current time	72	dbspaceobjectinfo, sp_iq	33
current timestamp	72	dbspaces	29
current user	81	creating	30
current utc timestamp	73	default dbspace	31
cursors	6	information about	32
		rename	31
		dbstatistics, sp_iq	50
		dbstop (utility)	26
		dbtran (utility)	54
		ddl_options2 (option)	110
		debug_messages (option)	128
		declare	
		exception	129
		local temporary table	83
		variable	129
		default index	See FP index
		default IQ port number (2638)	17, 21
		default IQ server (-xd)	21, 59
		default username/password	27
		default value, resetting IQ options to	118
		default, for column	81
		default_dbspace (option)	31
		default_disk_striping (option)	31
data extraction (temp_extract)	112		
data federation (proxy tables)	34		
data loading into IQ	107		
data modeling	11		
database consistency checks	50		
database corruptions	50		
database properties	13		
database server window (.srvlog)	18		
database vs. server	10, 20		
datalength()	80		
datatypes			
conversion functions	71		
system	69		
user-defined	71		

default_having_selectivity_ppm (opt)	121	E	
default_kb_per_stripe (option)	31		early_predicate_execution (option).... 122
default_like_match_selectivity(option)	121		-ec (encrypted client/server conn.) 22
default_like_range_selectivity(option)	121		editions of IQ..... 14
degrees()	78		-ek (database encryption key) 22
delayed_commits (option)	115		electronic code samples..... 8
delete	102		embedded SQL..... 6
demo database (iqdemo)	17		empty string..... 67
dense_rank() (OLAP)	106		emptyfile, sp_iq..... 32
dependent_views, sa	88		encryption
deprecated, options/features	6		client-server connections..... 22
derived table	100		entire IQ database..... 27
describe (dbisql command)	62		table columns..... 87
-dh (hide db name)	24		endian..... 109, 113
difference()	80		eng_properties, sa..... 13
directI/O	32		Enterprise Edition of IQ..... 14
directory access			environment variables..... 15
filesystem access via proxy table	36		log settings..... 24
disable_ri_check (option)	85		-ep (database encryption key prompt)..... 22
discovery, of IQ servers	24		errata..... 8
dbisql	61		error log files..... 18
disk striping, across dbfiles	31		errmsgs()..... 130
disk_striping (option)	31		-es (unencrypted shared connections)..... 23
disk_striping_packed (option)	31		escape, with like operator..... 69
distinct (operator)	100		escape_character (option)..... 58, 68
in aggregate	103		estdbspaces, sp_iq..... 33
distributed query processing (DQP)	44		estjoin, sp_iq..... 94
dml_by_zero_error (option)	122		estspace, sp_iq..... 33
dml_options10 (option)	120		ETL (InfoPrimer)..... 11, 15, 107
domain..... See user-defined datatype			evaluation version of IQ..... 14
dow()	73		events..... 6
download, IQ software and ESDs	53		exception handler, in SQL..... 129
DQP (distributed query processing)	44		execute (procedure)..... 96
dqp_enabled			execute (SQL statements)..... 128
login policy..... 38			execute immediate (SQL statements)..... 128
option	45		exists (subquery operator)..... 101
drop connection	55		exit
drop database	28		dbisql command..... 62
drop dbspace	31		isql/iqisql command..... 65
drop externlogin	35		exp()..... 77
drop function	98		exp_weighted_avg() (aggregate)..... 106
drop join index	93		exp_reallpasswords, sp_..... 37
drop logical server	45		Express Edition of IQ..... 14
drop login policy	39		external login..... 35
drop message	130		external UDFs..... 99
drop multiplex server	43		Java, Perl, etc..... 99
drop procedure	96		scalar UDFs..... 99
drop server	35		table parameterized functions..... 99
drop table	82		table UDFs..... 99
drop text configuration	95		user-defined aggregates..... 99
drop text index	95		extracting data from IQ..... 112
drop text user	37		
drop variable	129		F
drop view	88		false (logical constant)..... 69
dropgroup, sp_	39		federation (proxy tables)..... 34
dropleaks (sp_iqcheckdb)	51		file DSN, example..... 64
droprologin, sp_	37		file, sp_iq..... 32
droprologin, sp_iq	37		filesystem access via proxy table..... 36
droptype, sp_	71		-fips (FIPS encryption)..... 23
dropuser, sp_	37		first_day_of_week (option)..... 74
dscp (OpenClient utility)	65		first_value() (aggregate)..... 106
dsedit (OpenClient utility)	65		flat FP index..... 88
DSN (Data Source Name, ODBC)	64		floor()..... 76
File DSN, example	64		for (loop, cursor)..... 128
-dt (temporary SA files directory)	24		force_drop (option)..... 51
DTTM index	90, 91		forced recovery (-iqfrec)..... 51
dummy, iq_ (table)	125		forcing an index..... 102
			foreign key...references (constraint)..... 85
			formatting date & time data..... 75

formatting query results as xml, html ... 63
 forward to (CIS, remote server) 35
 FP index (flat FP, FP1, FP2, FP3) 88
 FP index rollover 89
 setting rollover point 89
 fp_lookup_size (option) 89
 fp_lookup_size_ppm (option) 89
 frame (in over clause) 104
 free version of IQ 14
 freelist 34
 from (clause) 100
 full text search 94
 fuzzy search (full text search) 94, 95

G

-ga (automatic shutdown) 25
 -gc (checkpoint interval) 24
 -gd (start/stop db permission) 22
 get_request_profile, sa_ 19
 get_request_times, sa_ 19
 get_user_status, sa_ 37
 getdate() 72
 getinfo.sh/.bat (utility) 56
 -gk (stop IQ permission) 22
 -gl (load table permission) 22
 global temporary tables 83
 -gm (#connections) 21
 -gn (#threads) 23
 go (isql/iqisql command) 65
 goto 128
 -gp (catalog page size) 23
 grant
 authority 40
 connect (IQ 12.x) 55
 create on 41
 group 39
 membership in group 39
 permission 40
 with grant option 40
 graphical_plan() 120
 group by (clause) 103
 group_member() 39
 grouping(), with group by rollup/cube 103
 groups 39
 membership 39
 -gss (stack size) 23
 -gt (#CPUs) 22
 -gtc (#CPUs) 22
 -gu (create/drop db permission) 17, 22
 GUI mode (dbisql) 61
 GUID - newid() 77

H

Hadoop 99
 hash_pinnable_cache % (option) 123
 hash_thrashing_percent (option) 122
 having (clause) 103
 help, sp_iq 82, 88, 97, 98
 helpindex, sp_iq 92
 hextobigint() 72
 hextoint() 72
 HG index 90, 91
 HNG index 90, 91
 hour() 73
 hours() 73
 HTML
 formatting query results 63
 graphical query plan 119

html_plan() 120
 HTTP interface to IQ 6
 hyperthreading 130

I

identifiers 66
 quoted 66, 67
 using reserved words as 67
 identity column 86
 identity_enforce_uniqueness (option) 86
 identity_insert (option) 86
 if
 if-expression 127
 if-statement 127
 ifnull() 127
 ILM See Information Lifecycle Mgmt
 image (blob/long binary datatype) 69
 in
 logical operator 69
 subquery operator 101
 in system (SAY tables) 82, 124
 in_subquery_preference (option) 121
 index type
 CMR 90, 91
 composite (HG, CMP) 90
 DATE 90, 91
 default index See FP index
 DTTM 90, 91
 FP (flat FP, FP1, FP2, FP3) 88
 HG 90, 91
 HNG 90, 91
 join index 93
 LF 90, 91
 overview/summary 90
 text 95
 TIME 90, 91
 unique (LF, HG) 90
 WD (word) 90
 index, force 102
 index, sp_iq 92
 index_advisor (option) 92
 index_advisor_max_rows (option) 92
 index_alt, sp_iq 92
 index_col() 81
 index_preference (option) 120
 indexadvicex, sp_iq 92
 indexfragmentation, sp_iq 93
 indexinfo, sp_iq 33
 indexing recommendations 90
 indexmetadata, sp_iq 92
 indexsize, sp_iq 92
 indexuse, sp_iq 52
 InfoPrimer (ETL) 11, 15, 107
 Information Lifecycle Management 83
 inline view See derived table
 inner join 101
 input (dbisql command) 62
 insert...location 111
 insert...select 102
 insert...values 102
 insertstr() 78
 installer
 issues 54
 silent install 54
 integrated logins 6
 Interactive SQL See dbisql
 interfaces file (OpenClient) 65
 internationalization
 list character sets 29

vs. Native IQ connections 57
 operators (numeric,string,logical,etc.) . 68
 optimized FP index 89
 optimizer hints 102
 options
 affecting query execution 122
 affecting query plans 120
 affecting query results 121
 dbisql-specific options 63
 generating query plans 119
 IQ options, setting 117
 settings for native IQ or TDS conn. 58
 watch list 119
 or (logical operator) 68
 order by (clause) 100
 -os (max .srvlog size) 18
 os_file_cache_buffering (option) 32
 os_file_cache_buffering_tempdb (opt) 32
 outer join 101
 output (dbisql command) 62
 output redirection in dbisql/isql 113
 over (clause) 103

P

-p (network packet size) 22
 page size
 catalog store 27
 IQ store 27
 parallel execution
 backup devices 46
 index creation 90
 load table 109
 parameters (dbisql command) 62
 partial-width loading 109
 partitions
 table partitions 83
 window partitions (OLAP) 103
 password, change
 alter user 37
 sp_[iq]password 37
 password, default 27
 password, remove 37
 password, sp_[iq] 37
 password_expiry_on_next(login policy) 38
 password_grace_time (login policy) 38
 password_life_time (login policy) 38
 patindex() 78
 pattern matching 69, 78
 -pc (network compression) 22
 percent_rank() (OLAP) 106
 percentile_cont() (OLAP) 106
 percentile_dist() (OLAP) 106
 performance monitoring 52
 performance tips 130
 indexing 90
 load table 109
 Perl (API) 65
 permissions 40
 personal server (-gk all) 22
 PHP (API) 65
 physical I/O (sp_[q]status) 53
 pi() 77
 pkeys, sp_[iq] 86
 PlexQ See DQP
 port number
 default (2638) 21
 invalid or in use, at startup 56
 port number, default (2638) 17
 post_login_procedure (option) 39

power() 77
 PowerDesigner 11
 prejoin (join index) 93
 primary key (constraint) 85, 86
 print 128
 private inter-node comms (MPX) 43
 procedure, sp_[iq] 97, 98
 procedures, stored 96
 progress indication messages 18
 properties
 connection properties 13
 database properties 13
 server properties 13
 property() 13
 property_description() 12
 property_name() 12
 property_number() 12
 proxy procedure 36
 proxy tables 34
 creating, create [existing] table 35
 filesystem/directory access 36
 getting information about 36
 -pt (network compression) 22
 public (group) 39
 Python (API) 65

Q

-qi (IQ tray icon) 25
 -qs (startup errors) 25
 quarter() 73
 query optimizer hints 102
 query plan
 graphical (HTML) 119
 options for generating 119
 text 119
 query plan viewer (dbisql) 120
 query_detail (option) 119
 query_name (option) 120
 query_plan (option) 119
 query_plan_after_run (option) 119
 query_plan_append_date_to_file(opt) 119
 query_plan_as_html (option) 119
 query_plan_as_html_directory (opt) .. 119
 query_plan_text_access (option) 120
 query_plan_text_caching (option) 120
 query_rows_returned_limit (option) .. 122
 query_temp_space_limit (option) 122
 query_timing (option) 119
 quit (dbisql command) 62
 quit (isql/iqisql command) 65
 quoted_identifier (option) 58, 66, 67
 -qw (IQ tray icon) 25

R

-r (readonly) 24
 radians() 78
 raiserror 130
 rand() 77
 random numbers 77
 rank() (OLAP) 106
 raw devices 30
 read (dbisql command) 62
 Real-Time Loading option 107
 rebuildindex, sp_[iq] 92
 recompile, create procedure .. with 96
 redirecting output in dbisql/isql 113
 references (foreign key constraint) 85
 referential integrity constraints 85

set temporary option	118	sp_iqindexfragmentation	93
as dbisql-specific command	63	sp_iqindexinfo	33
set transaction isolation level	115	sp_iqindexmetadata	92
setcompression, sp_iq	33	sp_iqindexsize	92
shared memory protocol	60	sp_iqindexuse	52
shared temp store (iq_shared_temp)	30	sp_iqjoinindex	94
shmem (shared memory protocol)	60	sp_iqjoinindexsize	94
unencrypted connections	23	sp_iqlmconfig	15
showcompression, sp_iq	33	sp_iqlocks	116
showpsexec, sp_iq	123	sp_iqmodifyadmin	39
shutting down IQ	25	sp_iqmpxcheckdqpconfig	45
sign()	76	sp_iqmpxdumptvlog	43
signal	129	sp_iqmpxfilestatus	43
silent install	54	sp_iqmpxinconnpoolinfo	44
similar()	80	sp_iqmpxinheartbeatinfo	44
simplex	41	sp_iqmpxinfo	43
sin()	77	sp_iqmpxvalidate	43
Single App Server Edition of IQ	14	sp_iqmpxversioninfo	117
single-node mode (MPX)	44	sp_iqobjectinfo	33
single-user mode	22	sp_iqpassword	37
sizing guide, for IQ	130, 131	sp_iqpkcs	86
Small Business Edition of IQ	14	sp_iqprocedure	97, 98
-sn (server name)	21	sp_iqrebuildindex	92
snapshot versioning	113, 117	sp_iqrelocate	31
SOAP interface to IQ	6	sp_iqrestoreaction	49
some (subquery operator)	101	sp_iqrowdensity	92
sort_pinnable_cache_percent(option)	124	sp_iqsetcompression	33
sortkey()	80	sp_iqshowcompression	33
soundex()	79	sp_iqshowpsexec	123
sp_addgroup	39	sp_iqspaceinfo	33
sp_addlogin	37	sp_iqspaceused	28
sp_addtype	71	sp_iqstatistics	53
sp_adduser	39	sp_iqstatus	21, 28, 49, 53, 117
sp_changegroup	39	sp_iqsysmon	52
sp_dropgroup	39	sp_iqtable	82
sp_droplogin	37	sp_iqtableuse	52
sp_droptype	71	sp_iqtransaction	117
sp_dropuser	37	sp_iqunusedcolumn	52
sp_expireallpasswords	37	sp_iqunusedindex	52
sp_iq_reset_identity	87	sp_iqunusedtable	52
sp_iqaddlogin	37	sp_iqversionuse	117
sp_iqbackupdetails	49	sp_iqview	88
sp_iqbackupsummary	49	sp_iqwho	51
sp_iqcardinality_analysis	94	sp_iqworkmon	52
sp_iqcheckdb		sp_login_environment	58
check consistency/corruptions	50	sp_password	37
repair corruptions	51	sp_remote_columns	35
resetlocks (MPX)	44	sp_remote_tables	35
sp_iqcheckoptions	19	sp_tsq_environment	58
sp_iqclient_lookup	52	space usage, monitoring	52
sp_iqcolumn	82	space()	79
sp_iqcolumnuse	52	spaceinfo, sp_iq	33
sp_iqconnection	51	spaceused, sp_iq	28
sp_iqconstraint	85	spatial data	70
sp_iqcontext	52	split partitions	84
sp_iqcopyloginpolicy	38	SQL Anywhere, inside IQ	124
sp_iqdbspace	32	SQL UDF	
sp_iqdbspaceinfo	32	scalar-valued	98
sp_iqdbspaceobjectinfo	33	table-valued	98
sp_iqdbstatistics	50	sql.ini file (OpenClient)	65
sp_iqdroplogin	37	sql_flagger_error_level (option)	126
sp_iqemptyfile	32	sql_flagger_warning_level (option)	126
sp_iquestdbspaces	33	sqlcode	130
sp_iquestjoin	94	sqldialect()	126
sp_iquestspace	33	sqlflagger()	126
sp_iqfile	32	sqlstate	130
sp_iqhelp	82, 88, 97, 98	sqrt()	77
sp_iqhelpindex	92	square()	77
sp_iqindex	92	srvlog file (.srvlog)	18
sp_iqindex_alt	92	SSL	23
sp_iqindexadvice	92	standard deviation	105

standby copy of IQ database.....	50	partitioned tables	83
start database.....	20	proxy tables	34
start/stop logging (dbisql command)....	62	SA/in system tables	82, 124
start_asiq.....	See start_iq	temporary tables(local,global,#tmp)	83
start_iq (utility).....	16	tableuse, sp_iq.....	52
startup command line	17	tan()	77
startup error message	55	TDS/OpenClient.....	See OpenClient/TDS
startup switches.....	21	temp cache size (-iqtc).....	21
multiplex.....	44	temp store (iq_system_temp).....	30
statistical aggregates.....	105	temp_extract_* (options).....	112
statistics, sp_iq.....	53	temp_extract_append.....	112
status, sp_iq.....	21, 28, 49, 53, 117	temp_extract_binary.....	112
stddev() (aggregate).....	105	temp_extract_column_delimiter.....	113
stddev_pop() (aggregate).....	105	temp_extract_directory.....	112
stddev_samp() (aggregate).....	105	temp_extract_name1..8.....	112
stderr file (.stderr).....	18	temp_extract_null_as_empty.....	113
stop database.....	21	temp_extract_null_as_zero.....	113
stop engine.....	25	temp_extract_quote.....	113
stop_asiq.....	See stop_iq	temp_extract_quotes.....	113
stop_iq (utility).....	25	temp_extract_quotes_all.....	113
stopping IQ.....	25	temp_extract_row_delimiter.....	112
storage, required for datatypes.....	69	temp_extract_size1..8.....	112
stored procedures	96	temp_extract_swap.....	112
call as function	98	temp_reserved_dspace_mb (option).....	31
str().....	80	temporary tables	83
str_replace().....	79	text (glob/long varchar datatype).....	69
string functions	78	text configuration object	94, 95
string operators	68	text index	94, 95
string().....	80	text search, full.....	94
string_truncation (option).....	68	text_delete_method (option).....	96
striping, across dbfiles	31	text_index_stats_sa.....	96
strtouid().....	77	text_index_vocab_sa.....	96
stuff().....	80	thrashing.....	122
-su (utility mode password).....	17, 22	ti (Client timeout).....	24
subquery_caching_preference (opt).....	121	TIME index.....	90, 91
subquery_flattening_percent (option).....	121	time series functions	107
subquery_flattening_preference (opt).....	120	time zones	70, 73
substr().....	79	time_format (option).....	58
substring().....	79	time_zone_adjustment (option).....	58, 73
substring64().....	79	timestamp.....	69
sum() (aggregate).....	104	timestamp_format (option).....	58
suppress_tds_debugging (option).....	58	tl (client liveness).....	24
suser_id().....	37, 81	TLS (Transport Layer Security).....	22
suser_name().....	37, 81	TLV log	43
switches, startup.....	21	to_char().....	29
multiplex.....	44	to_nchar().....	29
Sybase Central.....	54	today().....	72
Sybase Control Center (SCC).....	54	tools for the DBA.....	54
Sybase Replication Server.....	112	TPF.....	See table parameterized function
synchronize join index.....	93	-tq (automatic shutdown time).....	24
synchronizing nodes (MPX).....	42	traceback().....	130
syntax conventions used in this book.....	7	transaction isolation	
sys (group).....	39	transaction isolation level.....	115
SySAM (license manager).....	14	versioning	113, 117
sysam (utility).....	15	transaction log file.....	27
syslog, on Unix/Linux.....	25	transaction, sp_iq.....	117
sysmon, sp_iq.....	52	transactions	
system (catalog store).....	30	chained/unchained.....	114
system functions.....	80	commands for handling.....	114
		essentials.....	113
		examples	115
		nested.....	115
		transactsql().....	126
		Transport Layer Security (TLS).....	22
		trial version of IQ.....	14
table parameterized functions (TPF).....	99	triggers.....	82
table user-defined functions		trigonometric functions.....	77
external UDFs.....	99	trim().....	79
SQL UDFs.....	98	troubleshooting	55
table, sp_iq.....	82	installation/installer.....	54
table-level versioning.....	113, 117	license keys	15
tables.....	81		
base tables.....	81		

locking yourself out 56
 startup 24
 true (logical constant) 69
 truncate table 84, 102
 truncate() 77
 truncation_length (dbisql option) 63
 truncnum() 77
 T-SQL, vs. Watcom SQL 125
 tsql_environment, sp_ 58
 tsql_variables (option) 58
 TUDF See table user-defined function

U

ucase() 79
 UDD See user-defined datatype
 UDF See user-defined functions
 -uf (fatal error) 25
 unchained transaction mode 114
 undocumented commands, warning & disclaimer 8
 unicode() 29
 Unicode, UTF8 29
 union 100
 unique (constraint) 85, 86
 unique index 90
 unistr() 29
 unknown (logical constant) 69
 unload 112
 unloading data from IQ 112
 unlocking a locked user 37, 38
 unserved license keys 15
 unstructured data 94
 unusedcolumn, sp_iq 52
 unusedindex, sp_iq 52
 unusedtable, sp_iq 52
 update 102
 upper() 79
 use 128
 user connections, number of 21
 user_id() 37, 81
 user_name() 37, 81
 user-defined datatypes 71
 user-defined functions 98
 external UDFs 99
 SQL UDFs, scalar-valued 98
 SQL UDFs, table-valued 98
 username/password, default 27
 users 37
 utc timestamp 73
 UTC, time zone 73
 util_db.ini 17
 utility mode (utility database) 17
 utility_db 17
 UUID - newid() 77
 uuidtostr() 77

var_pop() (aggregate) 105
 var_samp() (aggregate) 105
 varexists() 129
 variable assignment 129
 variables, create/declare 129

variance() (aggregate) 105
 verify_password_function (option) 38
 versioning 113, 117
 versions, getting details about 117
 versionuse, sp_iq 117
 view, sp_iq 88
 views 87
 virtual backup 48

W

waitfor delay 128
 waitfor time 128
 waiting for blocking locks 116
 watch list, options 119
 Watcom SQL, vs. T-SQL 125
 watcomsql() 126
 WD (word) index 90
 web services 6
 API 65
 weeks() 73
 weighted moving average (OLAP) 106
 weighted_avg() (aggregate) 106
 where (clause) 100
 while (loop) 127, 128
 who, sp_iq 51
 width_bucket() (OLAP) 106
 wildcard characters 69, 78
 window (clause) 103
 window frame (in over clause) 104
 window partition (OLAP) 103
 windowing queries (OLAP) 103
 Windows service, setup IQ as 19
 word (WD) index 90
 workload monitoring 52
 workmon, sp_iq 52

X

-x (TCP/IP ports) 21
 -xd (default IQ server) 21
 XML datatype 70
 formatting query results as 63

Y

year() 73
 years() 73
 ymd() 73

Z

-z (connection diagnostics) 18
 -ze (environment variables) 24
 zeros, padding with leading 80
 ZFS filesystem 30
 -zl (capture last statement) 24
 -zn (connection diagnostics) 19
 -zo (connection diagnostics) 19
 -zr (connection diagnostics) 19
 -zs (connection diagnostics) 19