solidDB
Getting Started Guide

Version 6.0 | June 2007

solidDB 6.0

solidDB Getting Started Guide

Copyright © 2007 Solid Information Technology Ltd.
Document number: SEGS60
Product version: 06.00.1011
Date: 12.06.2007

Table of Contents

List of Figures

List of Tables

Chapter 1. Welcome

solidDB provides the features you would expect to find in any industrial-strength database server — multithreaded architecture, stored procedures, and so on. Solid also provides other features, such as "hot standby" capability to provide protection against server failure, SmartFlow technology to distribute data among multiple nodes, and in-memory database capability to increase performance.

solidDB is available for on a very wide range of platforms, including platforms such as Windows, [1] Linux, Solaris, HP-UX and other UNIX platforms as well as Real-Time Operating Systems such as VxWorks and QNX.

1.1 About This Guide

solidDB Getting Started Guide introduces solidDB to you. This manual also gives you an overview of the other manuals in this set. We assume that when you start reading this manual you have already successfully installed the solidDB software package containing the database files, database drivers, and the product documentation. If you have not, do so now. For instructions on installation, see the Evaluation Setup Guide.

1.1.1 Organization

This guide contains the following chapters:

*   Chapter 2, *Introduction* describes the situation immediately after the solidDB installation. It also provides you with the background information necessary to administer and maintain solidDB in your network environment.

*   Chapter 3, *Preparing for the Evaluation* gives you instructions on running the license script for evaluating solidDB. It also briefly discusses the role and contents of the solidDB configuration file, `solid.ini`.

*   Chapter 4, *Starting solidDB and Creating Your First Database* explains how to start solidDB and how to create databases with it.

*   Chapter 5, *Connecting to solidDB* explains how to use the Solid SQL editor or SolidConsole to connect to the database.

*   Chapter 6, *Stopping and Restarting the Database* explains how to stop and restart solidDB.

*   Chapter 7, *Running Samples* provides an introduction on running samples provided in the solidDB distribution package.

---

[1]For a detailed list of supported Windows™ versions, please refer to Solid Release Notes.

- Chapter 8, *What Next?* explains how to continue evaluating solidDB.

Glossary

Glossary provides definitions of terms.

1.1.2 Audience

This guide is for Database Administrators commissioning solidDB for evaluation purposes.

1.2 Conventions

1.2.1 About solidDB

solidDB from Solid Information Technology (Solid) represents a family of advanced database solutions for mission-critical applications.

This documentation assumes that all options of solidDB are licensed for use. In some cases, however, a customer may choose not to license certain options. These include in-memory engine, disk-based engine, CarrierGrade Option (also known as "HotStandby" in previous releases), and SmartFlow Option. Please refer to your organization's contract with Solid, or contact your Solid account representative.

1.2.2 Typographic Conventions

This manual uses the following typographic conventions:

Table 1.1. Typographic Conventions

Format	Used for
Database table	This font is used for all ordinary text.
NOT NULL	Uppercase letters on this font indicate SQL keywords and macro names.
solid.ini	These fonts indicate file names and path expressions.
SET SYNC MASTER YES; COMMIT WORK;	This font is used for program code and program output. Example SQL statements also use this font.
run.sh	This font is used for sample command lines.
TRIG_COUNT()	This font is used for function names.

Format	Used for
`java.sql.Connection`	This font is used for interface names.
LockHashSize	This font is used for parameter names, function arguments, and Windows registry entries.
argument	Words emphasised like this indicate information that the user or the application must provide.
solidDB Administration Guide	This style is used for references to other documents, or chapters in the same document. New terms and emphasised issues are also written like this.
File path presentation	File paths are presented in the Unix format. The slash (/) character represents the installation root directory.
Operating systems	If documentation contains differences between operating systems, the Unix format is mentioned first. The Microsoft Windows format is mentioned in parentheses after the Unix format. Other operating systems are separately mentioned.

1.2.3 Syntax Notation

This manual uses the following syntax notation conventions:

Table 1.2. Syntax Notation Conventions

Format	Used for
`INSERT INTO` *table_name*	Syntax descriptions are on this font. Replaceable sections are on *this* font.
`solid.ini`	This font indicates file names and path expressions.
[ ]	Square brackets indicate optional items; if in bold text, brackets must be included in the syntax.
|	A vertical bar separates two mutually exclusive choices in a syntax line.
{ }	Curly brackets delimit a set of mutually exclusive choices in a syntax line; if in bold text, braces must be included in the syntax.

Format	Used for
...	An ellipsis indicates that arguments can be repeated several times.
. . .	A column of three dots indicates continuation of previous lines of code.

1.3 Solid Documentation

Below is a complete list of documents available for solidDB. Solid documentation is distributed in an electronic format, usually PDF files and web pages.

Electronic Documentation

- *Release Notes*. This file contains installation instructions and the most up-to-date information about the specific product version. This file (releasenotes.txt) is copied onto your system when you install the software.

- *solidDB Getting Started Guide*. This manual gives you an introduction to the solidDB.

- *solidDB SQL Guide*. This manual describes the SQL commands that solidDB supports. This manual also describes some of the system tables, system views, system stored procedures, etc. that the engine makes available to you. This manual contains some basic tutorial material on SQL for those readers who are not already familiar with SQL. Note that some specialized material is covered in other manuals. For example, the Solid "administrative commands" related to the High Availability (HotStandby) Option are described in the *solidDB High Availability User Guide*, not the solidDB SQL Guide.

- *solidDB Administration Guide*. This guide describes administrative procedures for solidDB servers. This manual includes configuration information. Note that some administrative commands use an SQL-like syntax and are documented in the *solidDB SQL Guide*.

- *solidDB Programmer Guide*. This guide explains in detail how to use features such as Solid Stored Procedure Language, triggers, events, and sequences. It also describes the interfaces (APIs and drivers) available for accessing solidDB and how to use them with a solidDB database.

- *solidDB In-Memory Database User Guide*. This manual describes how to use the in-memory database of solidDB In-memory Engine.

- *solidDB SmartFlow Data Replication Guide*. This guide describes how to use the Solid SmartFlow technology to synchronize data across multiple database servers.

- *solidDB AcceleratorLib User Guide*. Linking the client application directly to the server improves performance by eliminating network communication overhead. This guide describes how to use the AcceleratorLib library, a database engine library that can be linked directly to the client application.

  This manual also explains how to use two proprietary Application Programming Interfaces (APIs). The first API is the Solid SA interface, a low-level C-language interface that allows you to perform simple single-table operations (such as inserting a row in a table) quickly. The second API is SSC API, which allows your C-language program can control the behavior of the embedded (linked) database server

  This manual also explains how to set up a solidDB to run without a disk drive.

- *solidDB High Availability User Guide*. Solid CarrierGrade Option (formerly called the HotStandby Option) allows your system to maintain an identical copy of the database in a backup server or "secondary server". This secondary database server can continue working if the primary database server fails.

Chapter 2. Introduction

This chapter describes the situation immediately after the solidDB installation. It also provides you with the background information necessary to administer and maintain solidDB in your network environment.

2.1 System Architecture

solidDB uses a client/server model. In practice, the solidDB session consists of the following co-operating processes:

- The server process, which manages the database files, accepts connections to the database from client applications, and carries out actions on the database as requested by the clients.

- The client process, which is used to pass the required tasks (through the server process) to the database. There can be several client types: a client could be a command-line tool, a graphical application, or a database maintenance tool. Typically, different applications act as clients to connect to solidDB. There is also a graphical frontend application provided with the solidDB software package; the SolidConsole.

solidDB can also run within the application process. This is provided by solidDB AcceleratorLib. The AcceleratorLib is a function library that provides the same functionality and interfaces available with solidDB. A user application may be linked to this library. The linked application communicates with the server by using direct function calls, thus skipping the overhead required when the client and server communicate through network protocols such as the TCP/IP. Linking the application and server into a single executable provides higher performance. For more information, refer to *solidDB AcceleratorLib User Guide*.

The client and the server can be on different hosts, in which case they communicate over a network. solidDB provides simultaneous support for multiple network protocols and connection types. Both the database server and the client applications can be simultaneously connected to multiple sites using multiple different network protocols.

To submit a query (an SQL statement) to a database server, a client must be able to communicate with that database server. solidDB, like many other database servers, uses "drivers" to enable this communication. Client applications call functions in the driver, and the driver then handles the communications and other details with the server. For example, you might write a C program that calls functions in the Open Database Connectivity (ODBC) driver, or you might write a Java program that calls functions in the Java Database Connectivity (JDBC) driver.

For more information about the ODBC and JDBC drivers, and how to use them with your client applications, see the *solidDB Programmer Guide*.

2.2 solidDB Directory Structure

After the solidDB installation, you will have a new directory on your server called `solidDB 6.0`. The files and subdirectories in that directory are explained in the table below. For more detailed information, refer to the in-package documentation.

Table 2.1. solidDB Directory Structure

Location	Explanation
Root folder	The root folder contains, for example: • A script used to facilitate running samples in the database evaluation phase • The licence file • Open the `welcome.html` file in a browser for more information on solidDB.
bin	solidDB binary files.
manuals	solidDB documentation.
eval_kit/standalone	solidDB evaluation kit licence and initiation files. It will also hold your database once it is created.
include	C program headers.
jdbc	Java Database Connectivity (JDBC) API for solidDB.
lib	Program libraries.
odbc	Open DataBase Connectivity (ODBC) API for solidDB.
samples	Samples that can be used in the database evaluation phase and future application development.

2.3 Prerequisites

Make sure that you have a valid solidDB licence in the installation directory. The licence file name is `sol-id.lic`.

2.4 What Is in Your solidDB Development Kit

Your solidDB Development Kit (SDK) contains more than just a solidDB program. It also contains the solidDB AcceleratorLib library, sample client programs, and documentation.

2.4.1 Software

The solidDB Development Kit contains a complete set of solidDB software, including the AcceleratorLib linkable library, and various "utility" programs.

2.4.2 Samples

The SDK comes with several sample programs written in C, SQL, and Java to help you get started using the features in your solidDB.

2.4.3 Documentation

For a description of the manuals that come with solidDB, see Section 1.3, "Solid Documentation".

Chapter 3. Preparing for the Evaluation

3.1 Running the License Script

The evaluation license enables you to evaluate solidDB for a limited time, each time you create a new database. To use the evaluation product for a longer period of time, contact your Solid salesperson for a different license.

In order to be able to run samples, you must put a copy of the license file into each of the sample run directories. The easiest way to do this is to execute the script called

`copy_licenses` (`copy_licenses.bat` in Microsoft Windows)

This script copies the license file from the installation root directory to all appropriate directories.

3.2 Solid Configuration File

When you start solidDB, it reads configuration parameters from the `solid.ini` configuration file. You can also use solidDB without the configuration file, in which case the factory settings are used. The main difference between the factory settings and the configuration file is the database port number. In factory settings, the port number is *1964* whereas in the configuration file it is *1315*.

The Solid configuration file will serve most evaluation needs as long as solidDB is concerned. If there is a need to use other engines or options such as the carrier grade option, the configuration file must be modified accordingly. Samples of the corresponding engine options are included with the samples. In other words, you do not have to modify the `solid.ini` file (in the `/eval_kit/standalone` directory) to evaluate other Solid options.

The `solid.ini` configuration file specifies parameters that help customize and optimize solidDB. For example, the *FileSpec* parameter in the `solid.ini` configuration file specifies the directory and files names of the data fields in which the server stores the user data. Another parameter specifies the block size for the database.

You can view all parameter setting with the solsql command below. for instructions, see Section 5.1, "Using Solid SQL Editor".

ADMIN COMMAND 'par';

You can view section specific parameter settings with the solsql command:

```
ADMIN COMMAND 'par section_name';
```

You can also view parameters by using the SolidConsole. Navigate to Administration+Configuration→View. The parameter window is shown below:

Figure 3.1. SolidConsole Parameter Window
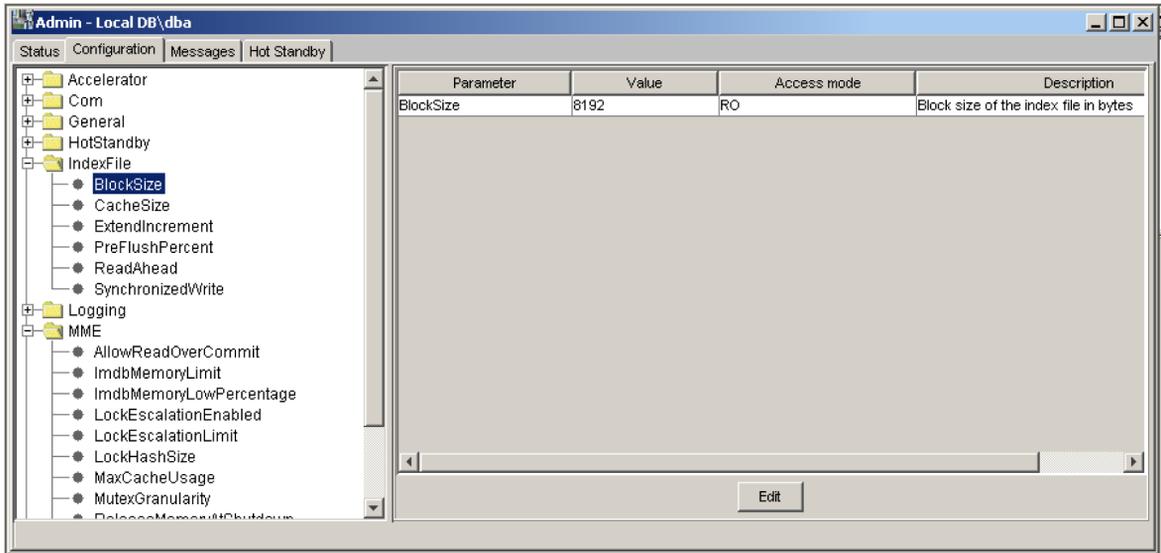
For more information, please refer to *solidDB Administration Guide*.

Chapter 4. Starting solidDB and Creating Your First Database

4.1 Starting solidDB

You can start solidDB as explained in the table below:

Table 4.1. Starting solidDB on Different Operating Systems

Operating System	To Start the Server...
Unix, Linux	Enter command **solid** at the command prompt. When you start the server for the first time, enter the command **solid -f** at the command prompt to force the server to run in the foreground.
Microsoft Windows	Click the icon labeled *solidDB Server* in the Start menu.

When you start solidDB, it checks if a database already exists. If no database is found, the engine automatically creates a new database in line with the solid.ini configuration file. However, if you want to create a database manually to a specific location, proceed as explained below.

4.2 Creating a Database Manually

To create a new database in the directory below:

\eval_kit\standalone

Open a command prompt window, go to the solidDB root directory and enter the following command:

bin\solid -c eval_kit\standalone

A window with the title "Creating a new database" will open. You will be asked to enter the system catalog name, username, and password.

◈ **Caution**

> There are no defaults for the username and password. You must remember the username and password to be able to access the database again.

Traditionally, Solid has used "dba" for the username and password of evaluation databases, since it is simple, easy to remember, and can be found in this document in case you forget it. To use it:

1. type any name you prefer as the catalog name

2. type "dba" as username and password

3. retype "dba" as the password

4. press OK

A new database is created and it is running.

4.3 Automatic Creation of a Database (Windows)

Start solidDB through the Start→Programs menu path. If no database is found, the engine automatically creates a new database in line with the `solid.ini` configuration file. In this case, the default name "dba" is used for the catalog, username and password.

A new database is created and it is running.

☞   **Note**

> Do not use these simple strings as username and password in the production environment.

4.4 Verifying the Database Installation

If you successfully created the database, your solidDB process is now running. The process name is solid (solid.exe in Microsoft Windows.

Chapter 5. Connecting to solidDB

After starting solidDB and creating your first database, you can test the configuration by connecting to the server from your workstation by using either the Solid SQL Editor or the SolidConsole.

If you have problems in connecting to the database with these instructions or if any errors should occur, please refer to *solidDB Administration Guide*.

5.1 Using Solid SQL Editor

With Solid SQL Editor, SQL statements (including the SQL ADMIN COMMANDs) can be issued at the command line, command prompt, or by executing a script file that contains the SQL statements.

5.1.1 Starting Solid SQL Editor

To start the Solid SQL Editor, enter the start command at your operating system prompt. The command syntax is as follows:

```
solsql "networkname" [userid [password]]
```

For example:

solsql "tcp hobbes 1315" dba dba

Or, if the server runs in the same computer:

solsql "tcp 1315" dba dba

When using Microsoft Windows, start the Solid SQL Editor from the icon in the Start menu

5.1.2 Connecting to Database

If you entered a valid user name and password when starting the Solid SQL Editor, you are connected to the database. If you did not give your user name and password when starting the Solid SQL Editor, you will be prompted for them. Enter your username and password (for example "dba", "dba") in the Solid SQL Editor.

The figure below shows the Solid SQL Editor after a successful connection to the database.

☞ **Note**

It can take a while before the connection can be established. Even though you can use the editor during the connecting phase, do not give any commands before you see a screen such as the one in the figure below.

Figure 5.1. Solid SQL Editor Connected
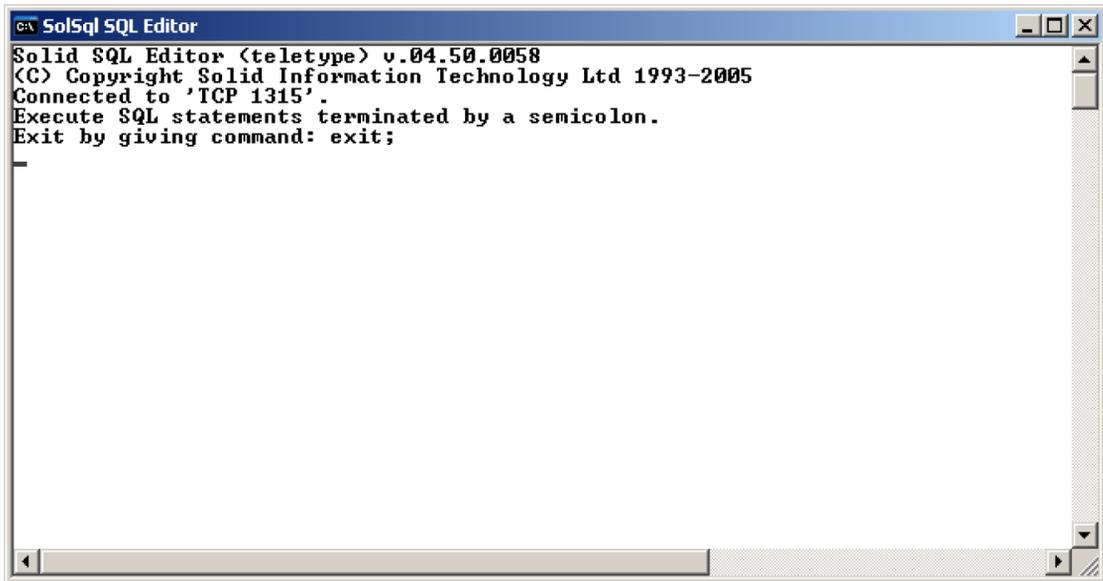
5.1.3 Shutting Down Solid SQL Editor

In all environments, you can close the solsql editor by entering the command:

exit;

5.1.4 Using Help

The Solid SQL Editor contains more detailed instructions for using the SQL commands. For a list of available commands, enter the command below:

ADMIN COMMAND 'help';

5.2 Using SolidConsole

SolidConsole is an easy-to-use graphical user interface for administering and monitoring solidDB data management engines and executing SQL queries and commands. SolidConsole program is not provided as part of the evaluation package. Instead, you must download it separately from the Solid web site:

http://www.solidDB.com/downloads

Download and install SolidConsole. We also recommend that you read the release notes.

5.2.1 Starting SolidConsole

To start SolidConsole, enter the following command at your operating system prompt:

java -classpath .\solconsole.jar;.\SolidDriver2.0.jar;. solconsole

or when using Microsoft Windows™, start SolidConsole from the icon in the Start menu

In Linux, it can also be launched by entering

run.sh

in the SolidConsole working directory.

5.2.2 Connecting to Database

When you open SolidConsole, it is disconnected from the database. To connect to the database, select Database→Connect from the menu, or use the **Ctrl**-**Shift**-**N** keyboard shortcut. The login screen with a local connection preconfigured is opened.

Figure 5.2. SolidConsole Login Screen
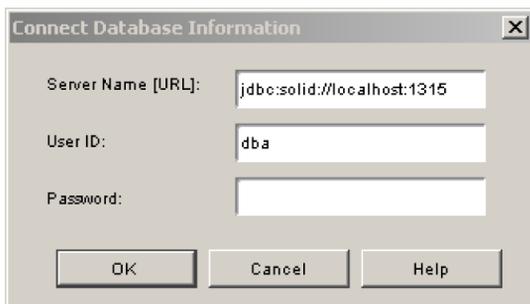
Enter your username and your password and press Enter to connect to the database.

5.2.3 Shutting Down SolidConsole

You can close the SolidConsole selecting File→Close.

5.2.4 Using Help

The SolidConsole contains more detailed instructions for using the SolidConsole. To open the help, select Help→Contents from the menu, or use the **F1** function button.

5.3 Entering Queries

Enter an SQL query, for example:

select table_name from tables;

In solsql, press Enter to execute the query. In SolidConsole, press **F6** to execute the query.

☞　**Note**

　　Notice the use of semicolon at the end of the SQL command.

For any solidDB, you should receive a resultset of 79 rows. See below for an example of the SolidConsole printout:

Figure 5.3. SolidConsole Printout
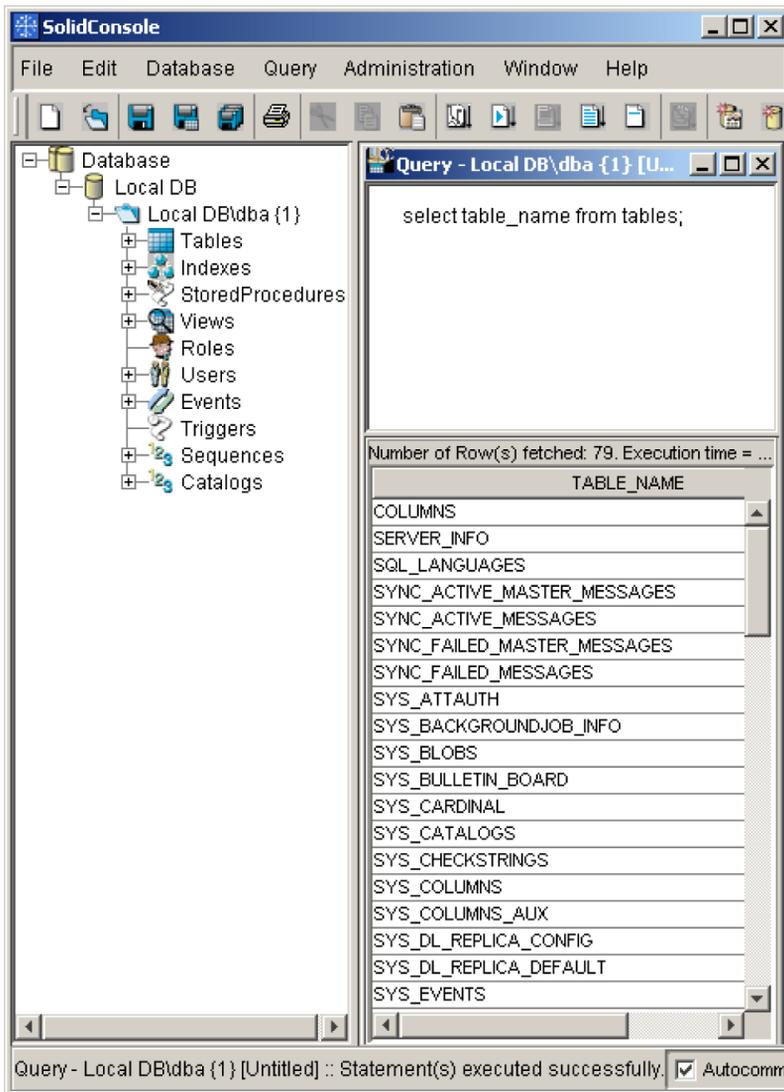
5.4 Checking the Database Status

The general server status may be retrieved by using the following command in the Solid SQL Editor:

ADMIN COMMAND 'status';

An example output from the solsql is shown below:

Figure 5.4. Example Output from Solsql

```
SolSql SQL Editor                                                          _ □ ×
ADMIN COMMAND 'status';
        RC TEXT
        -- ----
         0 Solid BoostEngine started at Fri Feb 03 13:58:35 2006
         0 Current directory is F:\Program Files\Solid\DatabaseEngine4.5\eval_kit\standalone
         0 Using configuration file F:\Program Files\Solid\DatabaseEngine4.5\eval_kit\standalone\solid.ini
         0 Memory statistics:
         0      38499 kilobytes
         0 Transaction count statistics:
         0      Commit Abort Rollback    Total Read-only  Trxbuf  Active Validate
         0         138     0       26      164      164       0       1       0
         0 Cache count statistics:
         0      Hit rate     Find      Read     Write
         0      96.8         1058      34        0
         0 Database statistics:
         0      Index writes          0  After last merge      0
         0      Log writes            0  After last cp         0
         0      Active searches       0  Average               1
         0      Database size      4032 kilobytes
         0      Log size            544 kilobytes
         0 User count statistics:
         0      Current Maximum Total
         0         1       1     1
20 rows fetched.
```

Chapter 6. Stopping and Restarting the Database

This section gives you instructions on stopping and restarting solidDB.

6.1 Stopping the Database with Solid SQL Editor

You can stop the database by using the Solid SQL Editor. Give the commands below. Use the semicolon at the end of the commands, and note that you must use single quotes, not double quotes.

admin command 'close';

admin command 'throwout all';

admin command 'shutdown';

exit;

The first command prevents any additional users from connecting. The second command throws out all connected users (except the one who issued the command). The third command shuts down solidDB. The fourth command exits from the solsql tool.

The first three commands may be replaced with this one:

admin command 'throwout force';

⬦ **Caution**

> When you shut down the server, it breaks the connection to solsql, and your solsql may show an error message such as: 14519: The user was thrown out from the server; connection lost

6.2 Stopping the Database with SolidConsole

You can stop the database by using the SolidConsole. Navigate to Administration→Status. Proceed as follows:

1. Press the Close button. This prevents any new users from connecting.

2. Press the Users button. This button opens up a list of current users. Press the Throw All button. This action throws out all connected users (except the one who issued the command).

3.   Press the Shutdown button.

The system asks for confirmation before performing the steps.

6.3 Restarting the Database

You can restart the database engine by moving to the solidDB root directory and entering the following command:

./bin/solid -c ./eval_kit/standalone

If you have the Solid "bin" directory in your path, you can go to the `eval_kit/standalone` directory to make it your current working directory, and start solidDB just by executing the command

solid

In Windows, to restart the evaluation database, use the Start menu

Chapter 7. Running Samples

You will find sample programs and scripts in the samples directory below the solidDB installation directory. There is a readme file in each samples subdirectory. Note that some sample scripts may remove and re-create database files residing in sample subdirectories. Only the database in the "standalone" directory is always left intact.

For more information, refer to the sample documentation provided in the solidDB distribution package.

Chapter 8. What Next?

We recommend that you read *solidDB Administration Guide*. This will help you configure, start, and maintain your database server.

We recommend that you try one of the sample programs provided with the solidDB Development Kit.

After you have read *solidDB Administration Guide*, you may want to read the guides that apply to any specific options that you plan to use:

- solidDB In-Memory Database User Guide

- solidDB High Availability User Guide

- solidDB AcceleratorLib User Guide

- solidDB SmartFlow Data Replication Guide

Most users will probably want to read the parts of the solidDB Programmer Guide that are relevant to them. If you will be writing client programs that query the server, you will need to read at least part of this manual.

Glossary

This glossary gives you a description of the terminology used in this guide.

A

Application Programming Interface (API)
An API is a way for one piece of code to use another piece of code. In the case of solidDB, client programs use an API (such as ODBC or JDBC) to communicate with the server and make requests of it. For example, your client application program may compose an SQL query and then use an ODBC or JDBC function call to send that query to the server for processing.

Solid also provides a proprietary database API called SA that can be accessed by using a special library.

C

Client/server computing
Client/server computing divides a large piece of software into modules that need not all be executed within the same memory space nor on the same processor. The calling module becomes the "client" that requests services, and the called module becomes the "server" that provides services. Client and server processes exchange information by sending messages through a computer network. They may run on different hardware and software platforms as appropriate for their special functions.

Two basic client/server architecture types are called two-tier and three-tier application architectures.

Communication protocol
A communication protocol is a set of rules and conventions used in the communication between servers and clients. The server and client have to use the same communication protocol in order to establish a connection. TCP/IP is an example of a common communication protocol.

D

Database administrator
The database administrator is a person responsible for tasks such as:

- managing users, tables, and indices

- backing up data

- allocating disk space for the database files

Database Management System (DBMS)
A DBMS is a system that stores information in and retrieves information from a database. A DBMS typically consists of a database server, administration utilities, an application interface, and development tools.

Database procedures
See stored procedures.

I

Index
An index of records has an entry for each key field (for example, employee name, identification number, etc.) and the location of the record. Indexes are used to speed up access to tables. The database engine uses indexes to access the rows in a table directly. Without indexes, the engine would have to search the whole contents of a table to find the desired row. A single table can have more than one index; however, adding indexes does slow down write operations, such as inserts, deletes, and updates on that table. There are two kinds of indexes: non unique indexes and unique indexes. A unique index is an index where all key values are unique.

J

Java Database Connectivity (JDBC)
JDBC is an application programming interface for the Java programming language. It defines how a client may access a database. It provides methods for querying and updating data in a database.

O

Open Database Connectivity (ODBC)
ODBC is a programming interface standard for SQL database programs. solidDB offers a native ODBC programming interface.

Optimizer Hints
Optimizer hints (which are an extension of SQL) are directives specified through embedded pseudo comments within query statements. The Optimizer detects these directives or hints and bases its query execution plan accordingly. Optimizer hints allow applications to be optimized under various conditions to the data, query type, and the database. They not only provide solutions to performance problems occasionally encountered with queries, but shift control of response times from the system to the user.

R

Relational database management system (RDBMS)

solidDB is an RDBMS, which stores and retrieves information that is organized into two-dimensional tables. This name derives from the relational theory that formalizes the data manipulation requests as set operations and allows mathematical analysis of these sets. RDBMSs typically support the SQL language for data manipulation requests.

S

Structured Query Language (SQL)

SQL is a standardized query language designed for handling database requests and administration. The SQL syntax used in solidDB is based on the ANSI X3H2-1989 Level 2 standard including important ANSI X3H2-1992 (SQL-92) extensions. Refer to *solidDB Administration Guide* for a more formal definition of the syntax.

SQL Access Group Call Level Interface (SAG CLI)

SAG CLI is a programming interface standard that defines the functions that are used to submit dynamic SQL clauses to a database server for execution. The ODBC interface is also based on SAG CLI. The Solid SQL API conforms to the SAG CLI standard.

Stored procedures

Stored procedures allow programmers to split the application logic between the client and the server. These procedures are stored in the database, and they accept parameters in the activation call from the client application. This arrangement is used by intelligent transactions that are implemented with calls to stored procedures.

T

Triggers

Triggers are pieces of logic that solidDB automatically executes when a user attempts to change the data in a table. When a user modifies data within the table, the trigger that corresponds to the command (such as insert, delete, or update) is activated.

Index

A
AcceleratorLib, 7, 9

C
Client/server model, 7
    client, 7
    server, 7

G
Getting started with solidDB, 7

J
JDBC, 7

L
Licence script, 11
Licencing, 11
    Licence script, 11

O
ODBC, 7

P
Preparing for the Evaluation, 11

S
Solid Configuration File, 11
solid.ini, 11
solidDB Directory Structure, 8

T
TCP/IP, 7