UNICOM® Global

solidDB
Fundamentals & Features

UNICOM® Global
Putting IT All Together.℠

Relational Database Software Powers Enterprise Applications

ERP

- General Ledger, Cash Management, Accounts Payable, Accounts Receivable, Fixed Assets, Human Resources, Payroll

CRM

- Sales and Marketing, Commissions
- Service
- Customer Contact and Call Center support

Data Warehousing

- Canned reports
- Ad-hoc Reporting
- OLAP
- Data Mining

Leading Relational Databases Efficiently Support

- **100s to 1,000s** of users
- **Milliseconds to seconds** response times
- **1,000s** of transactions **per minute**

As Number of Users Increase and Data Volumes Grow Data Management Performance Must Increase 10x

Communications

Financial Services

Web 2.0

Online Charging
- Authenticate and authorize
- Initiate service
- Manage credit balance
- Manage volume discounts

Brokerage Application
- Receive market feed
- Evaluate equity positions
- Check for fraud

Online Retail Web Site
- Authenticate user
- Manage personal wishlists
- Generate page contents with cross-sell data

- 100,000s to **1,000,000s** of concurrent requests
- **10s of microseconds** for database calls

- Evaluate **30,000+ rules** on **500 trades per second** for 15 million trades per day

- Facebook: **10,000,000** concurrent sessions = two billion page views a day
- Wikipedia: 3000 page views a second and **25,000** SQL requests per second

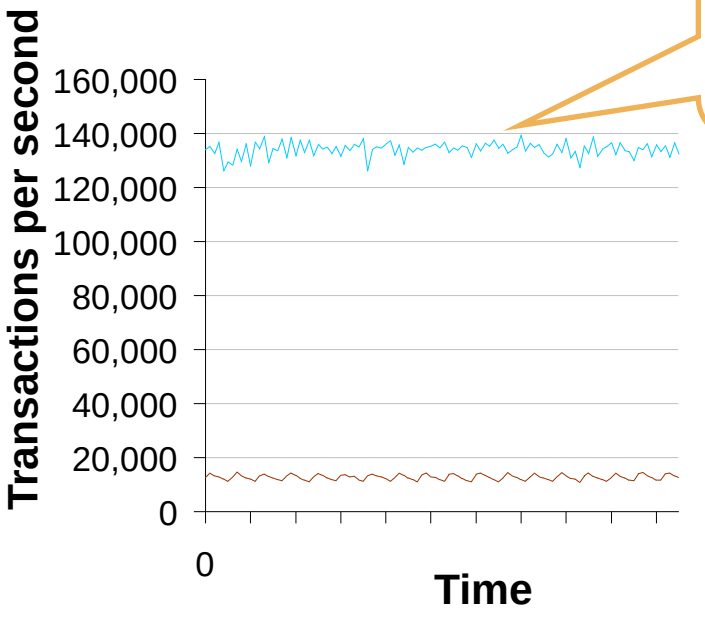
Comparison of On-Disk and In-Memory Databases

On-Disk Databases	In-Memory Databases
All data stored on disk, disk I/O needed to move data into main memory when needed	All data stored in main memory, no need to perform disk I/O to query or update data
Data is always persisted to disk	Data is persistent or volatile depending on the in-memory database product
Traditional data structures like B-Trees designed to store tables and indices efficiently on disk	Specialized data structures and index structures assume data is always in main memory
Support very broad set of workloads, i.e. OLTP, data warehousing, mixed workloads, etc.	Optimized for specialized workloads
Virtually unlimited database size (order of Terabytes, Petabytes)	Database size limited by the amount of main memory (Gigabytes)

Even when on-disk databases cache all data into main memory, in-memory databases **always** provide **shorter and more consistent response times** and **higher transaction throughput**

Relational, In-Memory, Database Technologies Process Performance Critical Data 10 times faster

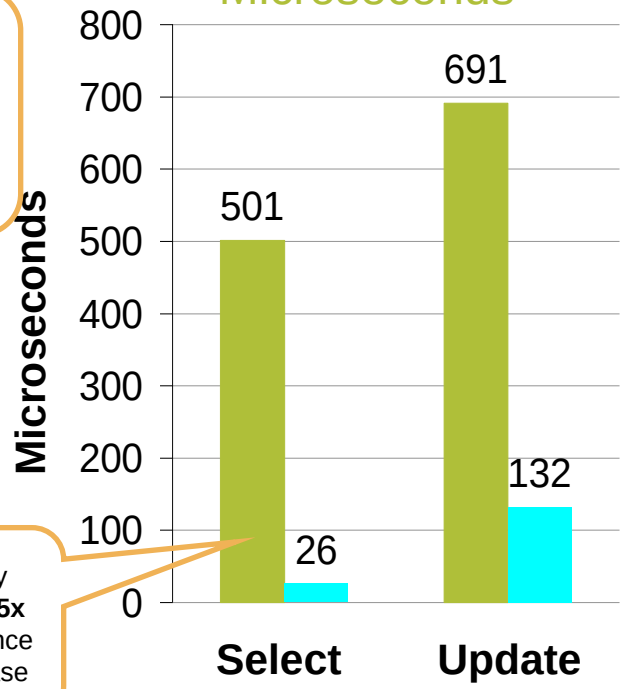
Throughput of Tens of Thousands of Transactions per Second



solidDB in-memory DB achieves **10x** the performance of an on-disk database even when on-disk database cached all data into main memory

- solidDB
- Disk-based database

Response Times Measured in Microseconds



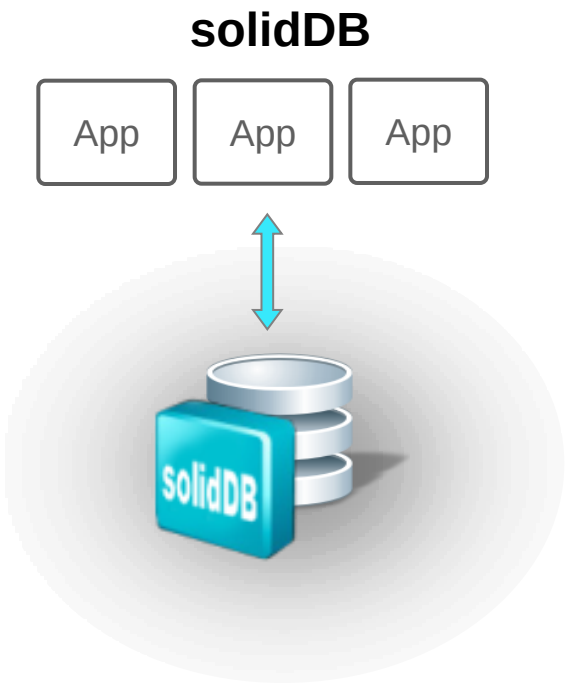
solidDB in-memory Cache achieves **5.25x to 20x** the performance of an on-disk database

- solidDB
- Disk-based database

Agenda

- Fundamentals of solidDB
- Installation and Basic Configuration of solidDB
- New features for solidDB 100

solidDB 100
Relational, In-Memory Database for Extreme Speed

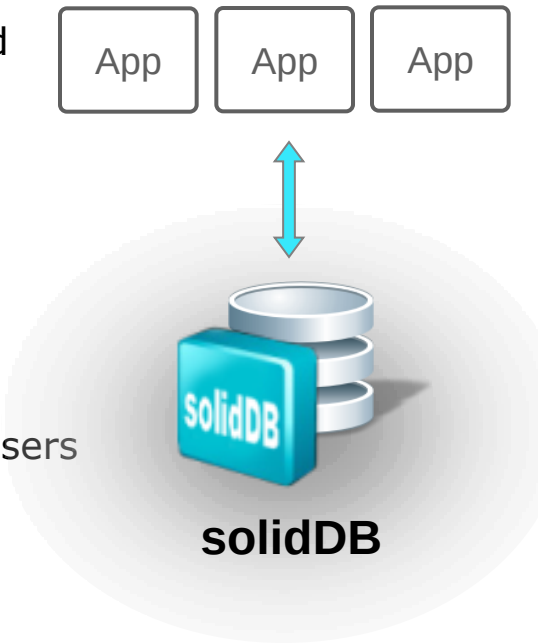


Platform Support

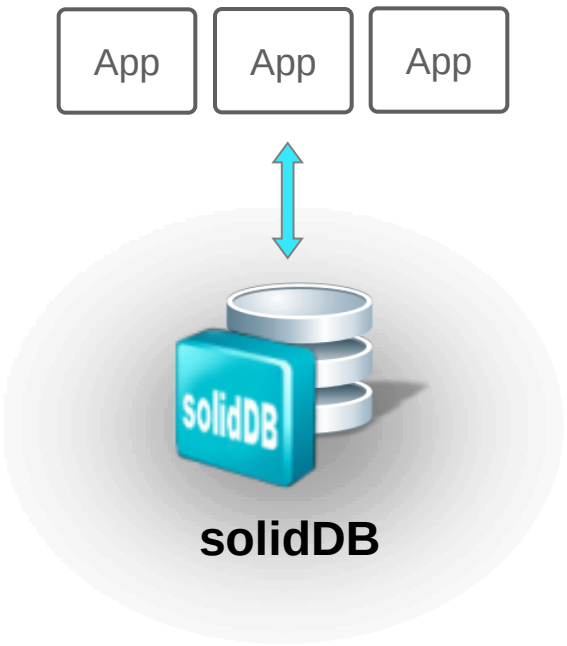
OS	OS Details	Hardware
AIX	AIX V5.3 / V6.1 / V7.1 for POWER5 / POWER6 / POWER7 systems	64-bit systems with POWER5™, or later processors are required
Linux	Red Hat Enterprise Linux (RHEL) 6 and 5 SUSE Linux Enterprise Server (SLES) 10 and 11	32-bit and 64-bit (x64) Linux: All systems based on Intel® or AMD processors that are capable of running the supported Linux operating systems (x86 and x64 systems)
Solaris	Solaris 10 for UltraSPARC and x86 servers	64-bit Systems with UltraSPARC or x86 processors are required
HP-UX	HP-UX 11i v2 and 11i v3 for HP 64-bit Integrity servers (Itanium-based systems)	Itanium-based 64-bit HP Integrity Series systems are required
Windows 	32-bit and 64-bit (x64) Windows Server 2008 and 2012, Standard, Enterprise and Datacenter Editions 32-bit and 64-bit (x64) Windows 7 and 8 Professional, Enterprise and Ultimate Editions 32-bit and 64-bit (x64) Windows Vista Business, Enterprise, and Ultimate editions	32-bit and 64-bit (x64) Windows: All systems based on Intel® or AMD processors that are capable of running the supported Windows (x86 and x64 systems)

solidDB

- In-Memory Relational Database
  - Extreme Speed
    - Designed to achieve very high throughput and very low response times (measured in microseconds)
    - Throughput of tens of thousand of transactions per second
    - Dual storage database
      - M-tables and D-tables with equal transactional capabilities
    - In-Memory tables keep data in main memory at all times
  - Extreme Availability
    - Supports 99.9999% availability
    - Provides instant application failover and transparency to users
  - Low Cost
    - Avoids costs associated with outages
    - Near-zero administration, runs virtually unattended
    - Easy to deploy, fully featured standards compliant relational database
    - Low development cost, leverages existing SQL skills

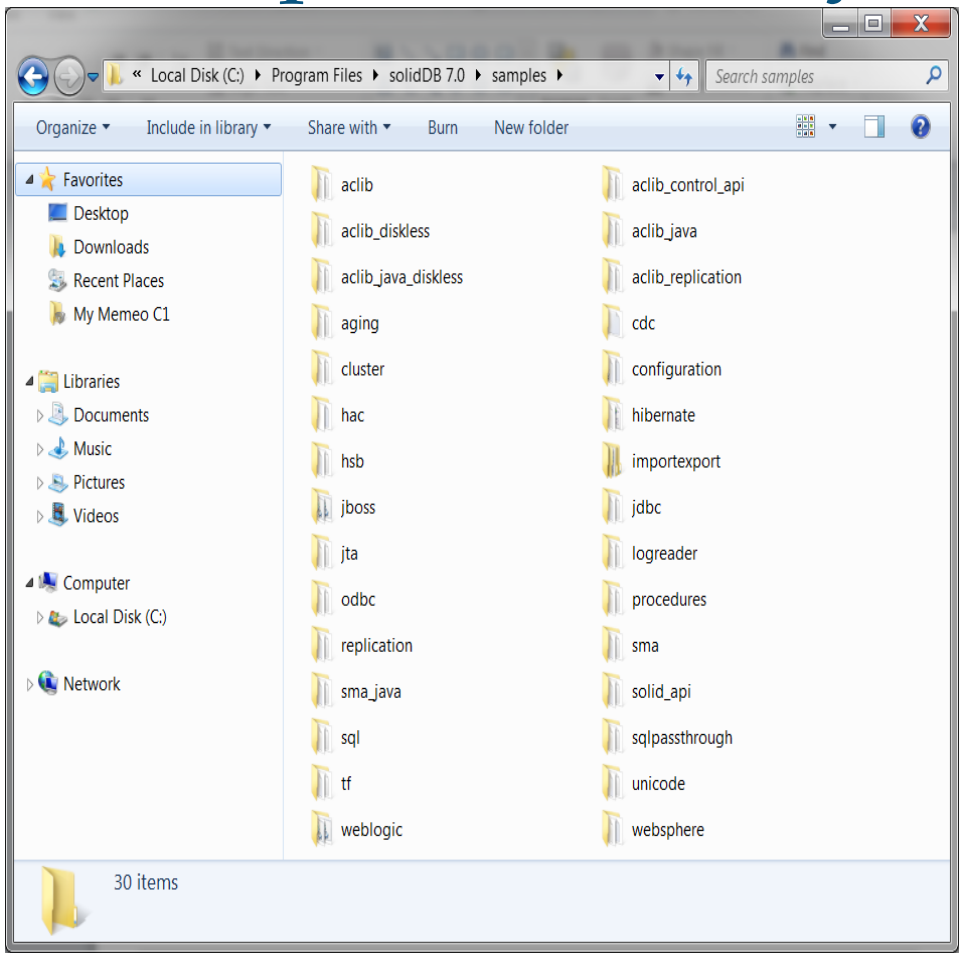


solidDB
In-Memory Relational Database



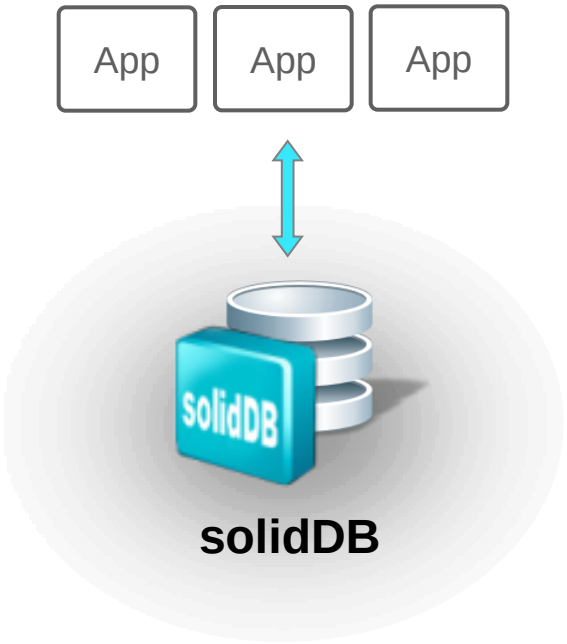
- Installation
  - solidDB is installed using Java based InstallAnywhere
  - Installation requires JVM in path
  - Evaluation License is provided in the 'base' directory
  - solidDB server is just one executable: ‘solid’
- Basic Configuration
  - solid.ini
- Operation
  - Startup
  - Shutdown
- Data Management Tools
  - solsql – command line tool for SQL
  - solcon – command line admin tool

solidDB Installation Samples Directory



- Each directory contains
  - readme.txt
  - makefile
  - runme.bat
  - Sample code
- Copy the solideval license file into the directory to execute the runme.bat
- Many of these samples have been used to create the bootcamp labs
  - Hot Standby
  - ODBC/JDBC
  - Data Aging
  - SQL Passthrough
  - Linked Libraries

solidDB Basic Configuration
solid.ini Configuration File

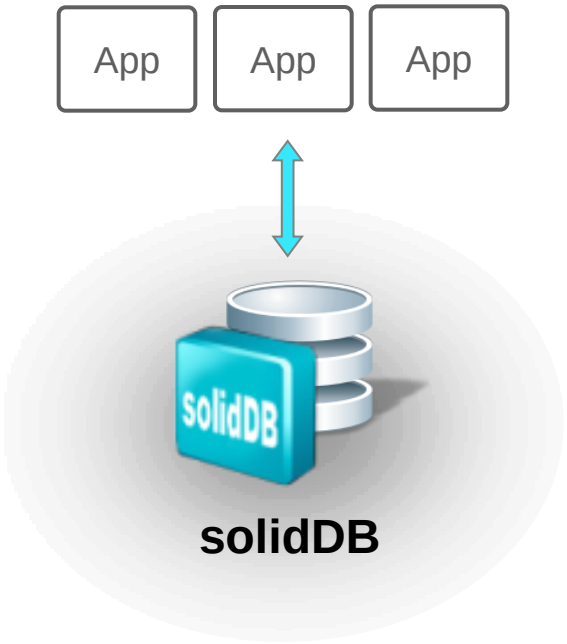


- Default Location is the current directory ('.')
  - Location can be specified with -c

    `-c /home/solid/db`
  - Location can be defined by SOLIDDIR environment variable
    - to override SOLIDDIR use

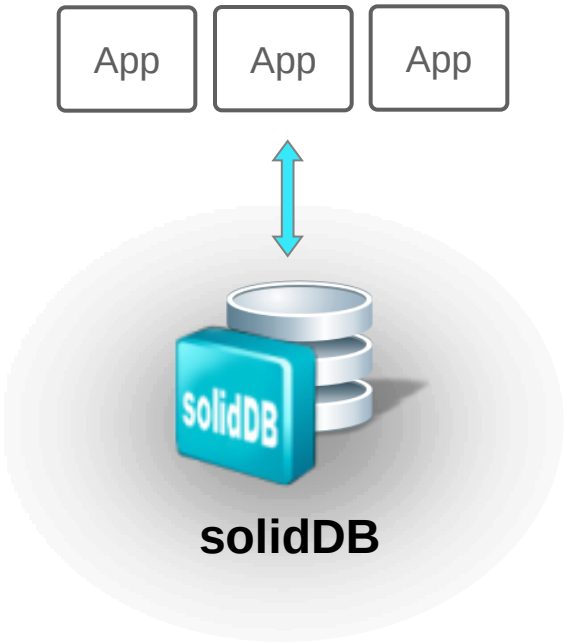
      `-x inifile:<path to solid.ini>`
- Define the way solidDB operates
  - Specify directories for the database, backup and log files
  - Set Communication settings
  - Define Memory usage
  - Create Timed commands
- Typically few parameters are needed
  - Default values apply in most cases

solidDB Basic Configuration
Setting Parameters in solid.ini



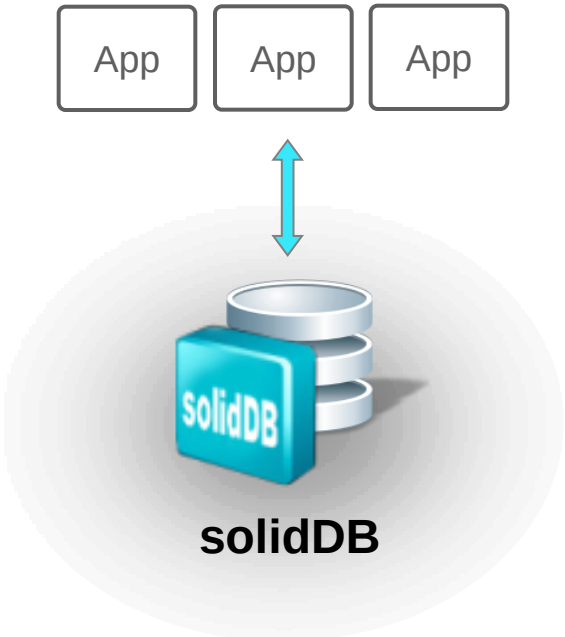
- Divided into sections delimited by square brackets
  - [Com], [General], [IndexFile], [Logging], [Srv], etc.
- Parameters syntax
  - <param_name>=<param_value>
- Spaces allowed
- Not case sensitive
- Comments follow semi-colon
  - ; This is a comment
  - <param_name>=<param_value> ; another comment
- Look for messages in solmsg.out

solidDB Basic Configuration
Access Modes



- Access mode indicates whether a parameter can be changed dynamically and when the change takes effect
  - RO: Read Only
    - The value cannot be changed dynamically
  - RW: Read/Write
    - The value may be changed dynamically and the change takes effect immediately
  - RW/Startup
    - The value may be changed dynamically but the change takes effect upon next server startup
  - RW/Create
    - The value can be changed but the change will not be applied until a new database is created

solidDB Basic Configuration Sample



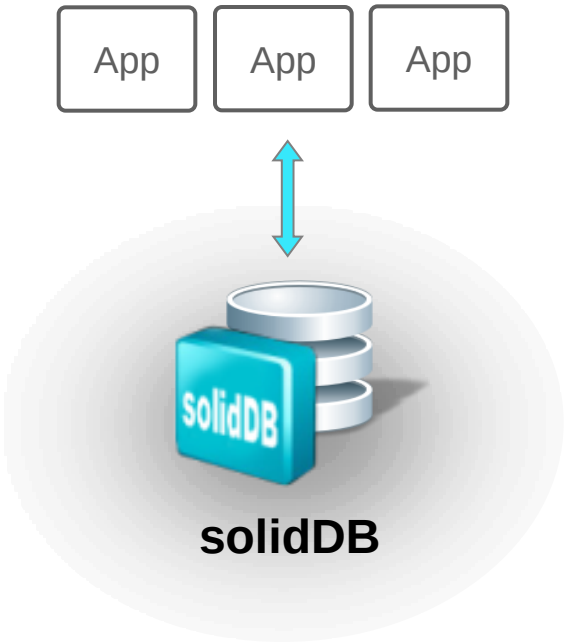

```
[Com]
Listen = tcpip 1315, shm solid1315

[Logging]
LogEnabled = NO

[Data Sources]
SOLDB = tcp 192.168.154.154 1315
```

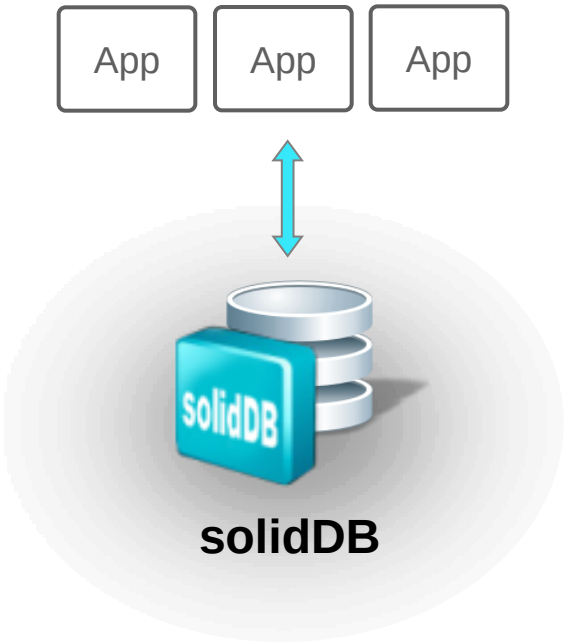
- [Com]
  - Listen
    - Defines Unique Network name (protocol + name)
    - Default: depends on OS
    - Access Mode = RW
- [Logging]
  - LogEnabled
    - Specifies whether to enable transaction logging
    - Default = Yes
    - Access Level = RW/Startup
- [Data Sources]
  - Purpose
    - Give the server a descriptive name
  - Definition
    - Logical Name
    - Network name

Data Management Tools



- solidDB *SQL Editor (solsql)*
  - Operations
    - All administrative operations: ADMIN COMMANDs
    - SQL statements
  - Use when
    - A command line based UI is needed
    - Executing SQL scripts
    - Testing simple SQL statements
  - Terminate command with semi-colon
  - Type `quit;` to exit
  - Options
    - Use **–a** for autocommit
    - Use **–t** to display execution times
- solidDB *Remote Control (solcon)*
  - Operations
    - Only ADMIN COMMANDs
  - Requires
    - SYS_ADMIN_ROLE or SYS_CONSOLE_ROLE
  - Use to
    - Speed up admin operations
    - Limit access to admin operations

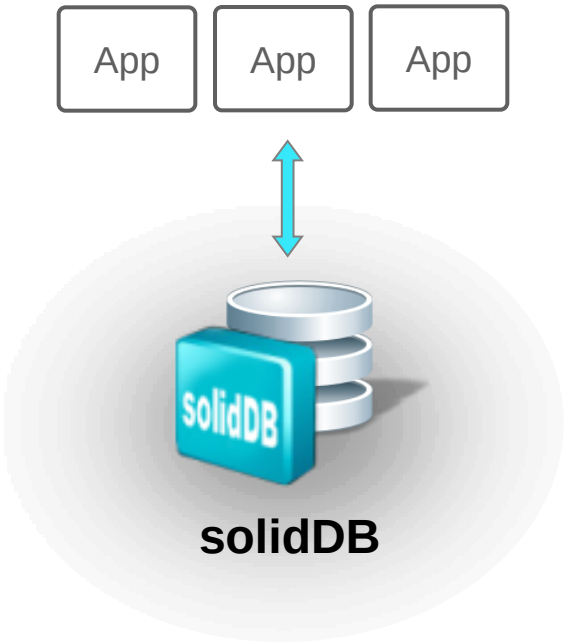
Operation
Starting solidDB



- Database is created automatically
- First time server is started requires
  - Default catalog name
  - System administrator name
  - System administrator password
- Special options use –x (partial list)
  - forcerecovery
  - execute: <input file>
  - reorganize
  - infodbfreefactor
  - ignoreerrors
  - keypwdfile:<filename>
  - testintegrity
  - Inifile:<full path to config file>

```
solid –c <working dir> –U username –P password –C catalog
```

Operation
Shutting down solidDB



- Abnormal shutdown doesn't corrupt database, but makes the next server start take longer
  - Power failures
  - System crashes
  - OS shutdown (sometimes)
- Ways to shut down the server gracefully
  - solid data management tools (solcon, solsql)
  - Server icon (Windows)
  - net stop (Windows system service)

```
ADMIN COMMAND 'shutdown';
ADMIN COMMAND 'shutdown force';

ADMIN COMMAND 'sd force';
```

solidDB 100 - What's New

- New Features and Improvements
  - Audit Information
    - Ability to monitor and log all the SQL activity in the database
  - Data Compression
    - Compression of disk-based table data
  - Cache Segment Partitioning
    - Ability to segment data to separate segments in the cache. E.g. hot and history.
  - Performance improvements
  - Better diagnostic functionality