IBM solidDB
Version 7.0

IBM solidDB Universal Cache Getting Started Guide

SC27-3848-05

Note
Before using this information and the product it supports, read the information in "Notices" on page 67.

First edition, fifth revision

This edition applies to V7.0 Fix Pack 8 of IBM solidDB (product number 5724-V17) and to all subsequent releases and modifications until otherwise indicated in new editions.

© Oy IBM Finland Ab 1993, 2013

Contents

Figures

Tables

Summary of changes

Changes for revision 05

- Section System requirements updated: support for Solaris 11 introduced in Fix Pack 6.

Changes for revision 04

- New section added:
  - Tutorial: Universal Cache evaluation setup

Changes for revision 03

- Information about supported Informix® editions added in section Supported backend data servers for Universal Cache.

  The following Informix editions are supported:
  - Informix Developer Edition
  - Informix Ultimate Edition
  - Informix Ultimate Warehouse Edition
- Section System requirements updated: support for Windows 8 and Windows Server 2012 introduced in Fix Pack 4.

Changes for revision 02

- Editorial corrections.

Changes for revision 01

- Section System requirements updated: support for System z® Linux introduced in Fix Pack 1.

About this manual

IBM® solidDB® Universal Cache is a solution for speeding up traditional disk-based SQL data servers by way of one or more solidDB in-memory database instances caching the data traffic between the applications and the data servers. IBM InfoSphere® Change Data Capture technology is used to implement the data replication between the solidDB and data server instances.

This guide provides an overview of the solidDB Universal Cache as well as instructions for installing and configuring the solidDB Universal Cache. Guidelines for handling failure and troubleshooting scenarios are also included. The chapter CDC for solidDB contains detailed instructions for how to install and configure the InfoSphere CDC for solidDB. This section is needed when configuring the solidDB Universal Cache; it provides comparable information to the *InfoSphere Change Data Capture, End-User Documentation* user manual for your backend data server.

This manual assumes that the reader has general database management system (DBMS) knowledge and familiarity with SQL and solidDB.

Typographic conventions

solidDB documentation uses the following typographic conventions:

Table 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.
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 emphasized like this indicate information that the user or the application must provide.
Administrator Guide	This style is used for references to other documents, or chapters in the same document. New terms and emphasized issues are also written like this.

Table 1. Typographic conventions (continued)

Format	Used for
File path presentation	Unless otherwise indicated, 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. There may also be different chapters for different operating systems.

Syntax notation conventions

solidDB documentation uses the following syntax notation conventions:

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

Part 1. Product overview

1 IBM solidDB Universal Cache

The Universal Cache capability in IBM solidDB is a solution for speeding up traditional disk-based databases. It incorporates a high-speed solidDB in-memory database that caches performance-critical data from a disk-based database. The in-memory cache allows the processing of the application load by a fraction of the response time it takes to be executed in the backend database. This improves performance, speed, and latency.

solidDB Universal Cache can be integrated with IBM DB2®, IBM Informix, Oracle, Microsoft SQL Server, and Sybase data servers. By combining the relational, in-memory data management capabilities of Universal Cache with the versatility of disk-based databases, applications benefit from the best of both worlds.

In addition to the solidDB in-memory database, the solution uses IBM InfoSphere Change Data Capture (referred to as InfoSphere CDC) technology for data replication. Both the solidDB cache and the backend database are self-contained database management systems (DBMSs).

1.1 Architecture and key components of Universal Cache

The architecture of the Universal Cache capability is based on three main components: the solidDB (in-memory) database (*cache*), a relational database server (*backend*), and the InfoSphere CDC data synchronization software that copies data to and from the cache and the backend.

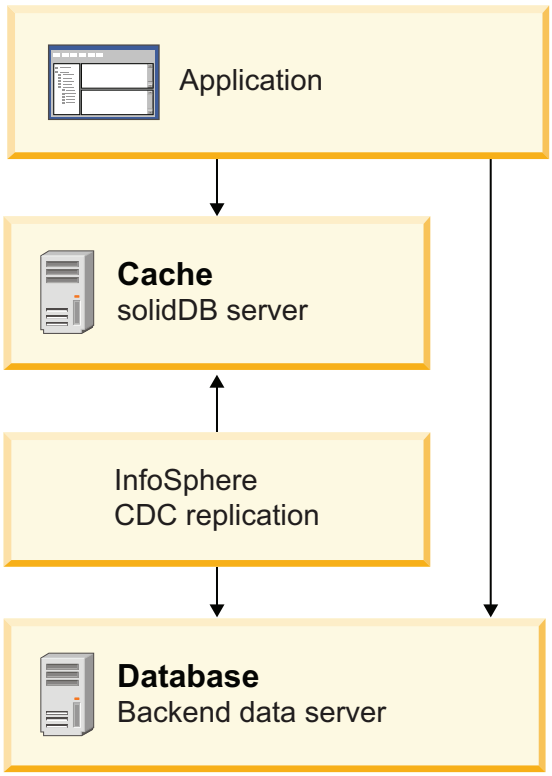

Figure 1. Architecture of Universal Cache

solidDB cache database

The solidDB server implements the cache database (or frontend) in the Universal Cache solution. The cache database benefits from various solidDB features, such as HotStandby that provides high availability and failover, or shared memory access (SMA) that enables collocating of data with the application.

Backend database

The backend database is a relational, disk-based data server that contains the data to be cached. In many Universal Cache deployment scenarios, you would have an existing database that you can simply replace with a cache database that sits between the backend database and application, making the database appear faster from an application perspective. Only minimal changes would be needed in the database interface layer.

InfoSphere CDC replication components

The IBM InfoSphere Change Data Capture replication software ensures that as changes are made to the cache database, the backend database is updated, and vice versa. The InfoSphere CDC replication software consists of the following components:

Replication engines

InfoSphere CDC replication engines are components that use log-scraping technologies, triggers, or both to capture changes between the two databases. The InfoSphere CDC replication engine for solidDB accesses the solidDB transaction log to capture data changes and transmits these changes to the backend replication engine, which copies the changes to the backend database.

Similarly, the InfoSphere CDC replication engine for the backend accesses the log (or uses triggers) to capture data changes in the backend and transmits these changes to the solidDB replication engine, which copies the changes to the backend database.

The replication engines run typically on the same hosts as the data servers.

Access Server

The InfoSphere CDC Access Server is a process that manages a Universal Cache deployment. It is typically executed as a daemon. The Universal Cache tooling communicate with the Access Server to allow deployments to be configured. Access Server also controls access to the replication environment; only users who have been granted the relevant rights can modify configurations.

Management Console

The InfoSphere CDC Management Console is an interactive GUI tool that you can use to configure and monitor replication (caching) subscriptions between the cache and backend databases.

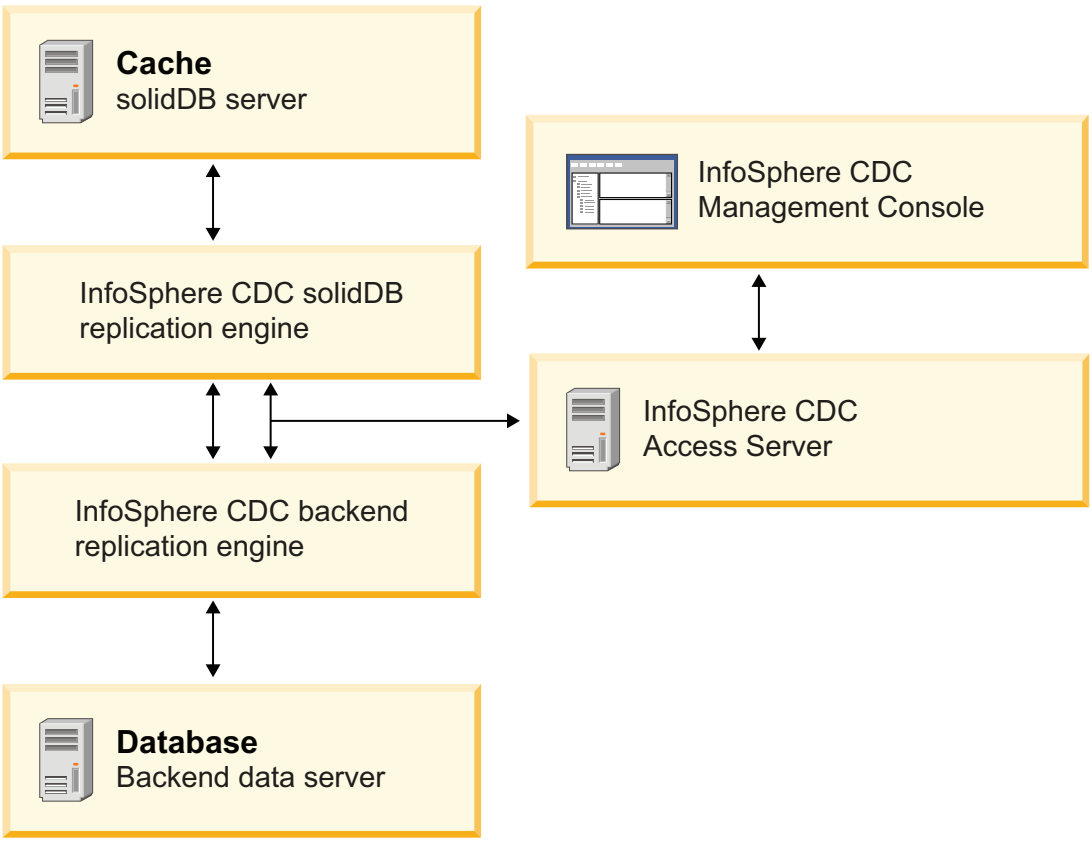

Figure 2. InfoSphere CDC replication components

SQL passthrough

The SQL passthrough functionality makes it possible to route SQL operations to the backend database, enabling applications to access data in both databases with a single interface.

For example, the Universal Cache system might be set up so that the frequently accessed data in table T1 is cached to the solidDB in-memory

database. The application can read and modify data in T1 - any changes are replicated to the backend database using InfoSphere CDC replication.

1.2 Universal Cache principles of operation

The solidDB database server maintains a cache database that contains a subset of data stored in the backend database. To use Universal Cache, you must first identify the data you want to cache and configure the environment accordingly. The data can then be loaded from the backend database to the cache, so that when applications run against the cache database, they can take advantage of high performance and low latency of the solidDB server. As changes are made to the data, the InfoSphere CDC replication technology synchronizes data between the cache database and the backend database.

The SQL passthrough functionality makes it possible to route SQL operations to the backend database, enabling applications to access data in both databases with a single interface.

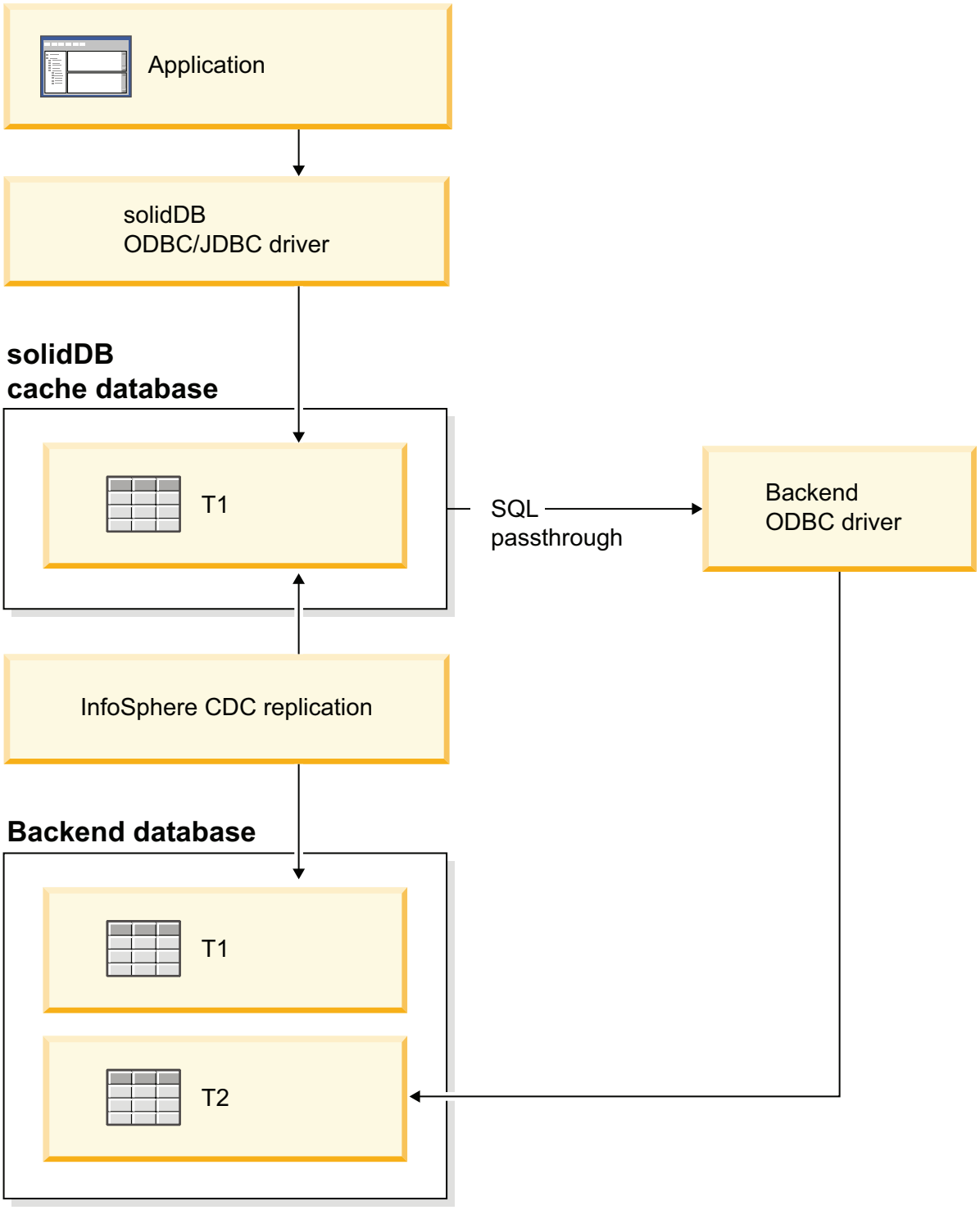

Figure 3. Universal Cache - principles of operation

For example, the Universal Cache system might be set up so that the frequently accessed data in table T1 is cached to the solidDB in-memory database. The application can read and modify data in T1 - any changes are replicated to the backend database using InfoSphere CDC replication.

Table T2 in the backend database contains data that the application needs to access only periodically. The Universal Cache can be configured so that when the

application needs to access data in T2, the queries to T2 are executed in the backend. A backend-specific ODBC driver is needed for the SQL passthrough connection.

1.3 Universal Cache features and functionality

Universal Cache is designed to meet a wide variety of different application and deployment needs; it can be configured as a read-only cache or as a read-write cache. Universal Cache also provides a powerful combination of replication models, schema mapping, and data transformation capabilities to move data between the cache and the backend database. The SQL passthrough functionality enables applications to access data both in the cache and the backend databases. SQL compatibility features ease the migration of applications from an enterprise data server to Universal Cache environment.

1.3.1 InfoSphere CDC replication in Universal Cache

Universal Cache includes an elaborate replication mechanism known as IBM InfoSphere Change Data Capture (InfoSphere CDC), available also as a separate product from IBM. The InfoSphere CDC components are responsible for carrying the data between the backend database and the cache database – both ways.

Replication with IBM InfoSphere Change Data Capture can take the form of one time *snapshot* (refresh) of cached data or continuous update propagation between the systems. The *continuous replication* method is asynchronous in nature: the data is first committed on the local system (the source) and then propagated to the other system (the target). The delay is usually within a second.

Log-scraping

InfoSphere CDC uses log-scraping technologies, triggers, or both to capture databases changes. The cache replication engine accesses the solidDB transaction log to capture data changes and transmits these changes to the backend replication engine, which copies the changes to the backend database. Similarly, the backend replication engine accesses the log (or uses triggers) to capture data changes in the backend database and transmits these changes to the cache replication engine, which copies the changes to the cache database.

Asynchronous replication considerations

Asynchronous replication method means that as applications write, for example, to the cache database, control is returned to the application as soon as the write completes; the application does not block, waiting for these updates to be successfully applied to the backend. Updates to the backend are not performed until the following tasks are completed:

1. The transaction has been committed in the cache database.
2. The entries for the transaction are scraped from the cache database log.

In the Universal Cache system, asynchronous replication benefits applications by reducing the round-trip time required to access data. Instead of potentially incurring an expensive network hop and writing to the backend database, applications can write directly to the solidDB database. Asynchronous replication means also that applications cannot assume that the backend database has been written to at the same time as the cache database, which can have ramifications for error recovery.

1.3.2 Read-only and read-write cache types

Depending on the application needs, Universal Cache can be deployed as a read-only cache or as a read-write cache.

Read-only cache

When configured as a read-only cache, the data is owned by the backend database. This ownership means that the data stored in the cache cannot be modified by the application. In read-only configuration, applications can modify the data directly in the backend database and changes can be synchronized to the cache, transaction by transaction, automatically or on-demand. This configuration is ideal for applications that require fast access to data that changes occasionally, such as price lists, or reference or lookup data.

Read-write cache

There are two deployment options for read-write cache, depending on the ownership of data.

Read-write cache with ownership at cache
When configured as a read-write cache where the data is owned by the cache, applications can read, add, modify, or delete data in the cache, but not in the backend database. Changes are propagated from the cache to the backend database, transaction by transaction, automatically, or on-demand. This configuration is useful for applications that have stringent service level agreements that demand short response times for a variety of data intensive operations.

Read-write cache with shared ownership
When configured as a read-write cache where the data ownership is shared, applications can update the same data in both the cache and in the backend database at the same time. In this case, changes to the data can be propagated automatically in both directions. Conflicts are detected and resolved by using predefined conflict resolution methods. This type of cache type is useful when applications need to update the data in the backend database while the data is also cached for read-write access.

1.3.3 SQL passthrough

The SQL passthrough feature enables applications to access data both in the frontend and backend data servers with a single connection. For example, SQL passthrough can be enabled in such a way that those SQL statements that cannot be executed in the solidDB frontend server are passed over to the backend. The SQL passthrough mode can be set per session or per transaction. By default, SQL passthrough is not enabled.

The connection between the frontend and backend is made with a backend compatible ODBC driver which is loaded dynamically in the solidDB server. The solidDB server uses this driver to execute the passthrough statements directly in the backend data server.

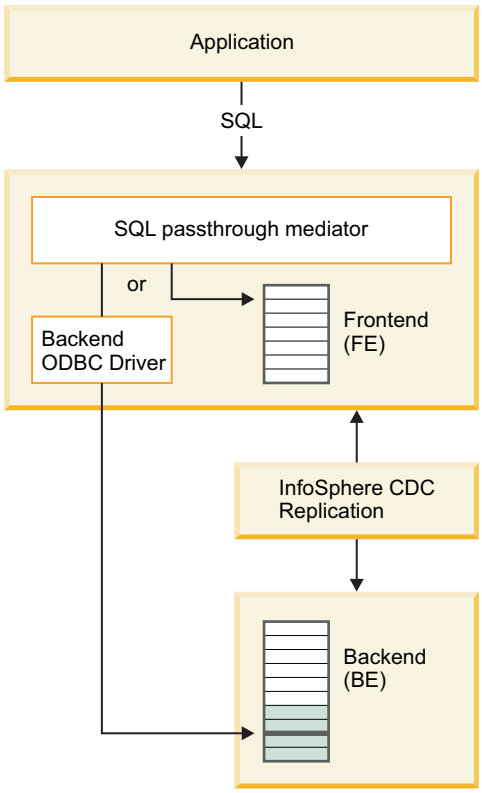

Figure 4. SQL passthrough

1.3.4 Schema mapping and data transformations

When configuring Universal Cache, you can use the data filtering and transformation features for defining exactly what data to cache.

Universal Cache enables you to, for example:

- Retain the same relational database schema as the backend database, or have a different schema
- Load the entire database or select just specific tables, columns, or rows

  For example, you can select individual columns to implement vertical partitioning of a database. Similarly, selecting only a subset of rows enables you to implement horizontal partitioning of a database.
- Extend the cache database schema with additional tables or derived fields
- Maintain the data format or apply data transformations such as data type conversions or summarizations

1.3.5 Collocating applications with the cache with SMA and LLA

You can magnify the advantage of a cache database by collocating your application with the cache database. By using shared memory access (SMA) or linked library access (LLA) , you can link the application with the solidDB server code and avoid any interprocess communications. With SMA and LLA, the resulting response times can be an order of magnitude (or more) shorter than those achieved with the network-based access to the backend database.

Disk-based databases are often accessed from separate client computers through a network, and data must be read from an external storage device (for example, a hard disk drive or a solid state drive) before it can be accessed by the application. Although advanced caching algorithms exist to store frequently used data in the database main memory (often referred to as a buffer pool), there is no guarantee that the requested data page will be available in the buffer pool at access time – therefore, a disk I/O operation is needed. Moreover, database durability requirements often dictate that log records are synchronously written to the storage device prior to any database updates being committed, thus introducing additional performance impact on the transaction response time seen by the application.

By using the SMA and LLA access methods available with solidDB, the Universal Cache system can collocate the data with the application. The combined cost of accessing data from the solidDB in-memory engine collocated with the application is significantly lower than accessing the data from the backend database server. All expensive access paths (network and synchronous disk access) can be removed.

SMA and LLA are implemented as library files that contain a complete copy of the solidDB server in a library form.

Shared memory access (SMA)
: With SMA, multiple applications can be linked to a dynamic driver library that contains the full database server functionality. This means that the applications ODBC or JDBC requests are processed almost fully in the application process space, without a need for a context switch among processes. To facilitate the processing of a common database, the driver has access to a shared memory segment initialized by the server.

Linked library access (LLA)
: With LLA, an application links to a static library or a dynamic library that contains the full database server functionality. This means solidDB runs in the same executable with the application, eliminating the need to transfer data through the network.

The SMA and LLA servers can also handle requests from remote applications which connect to the server through communications protocols such as TCP/IP. The remote applications see the SMA or LLA server as similar to any other solidDB server; the collocated applications see a faster, more precisely controllable version of the solidDB server.

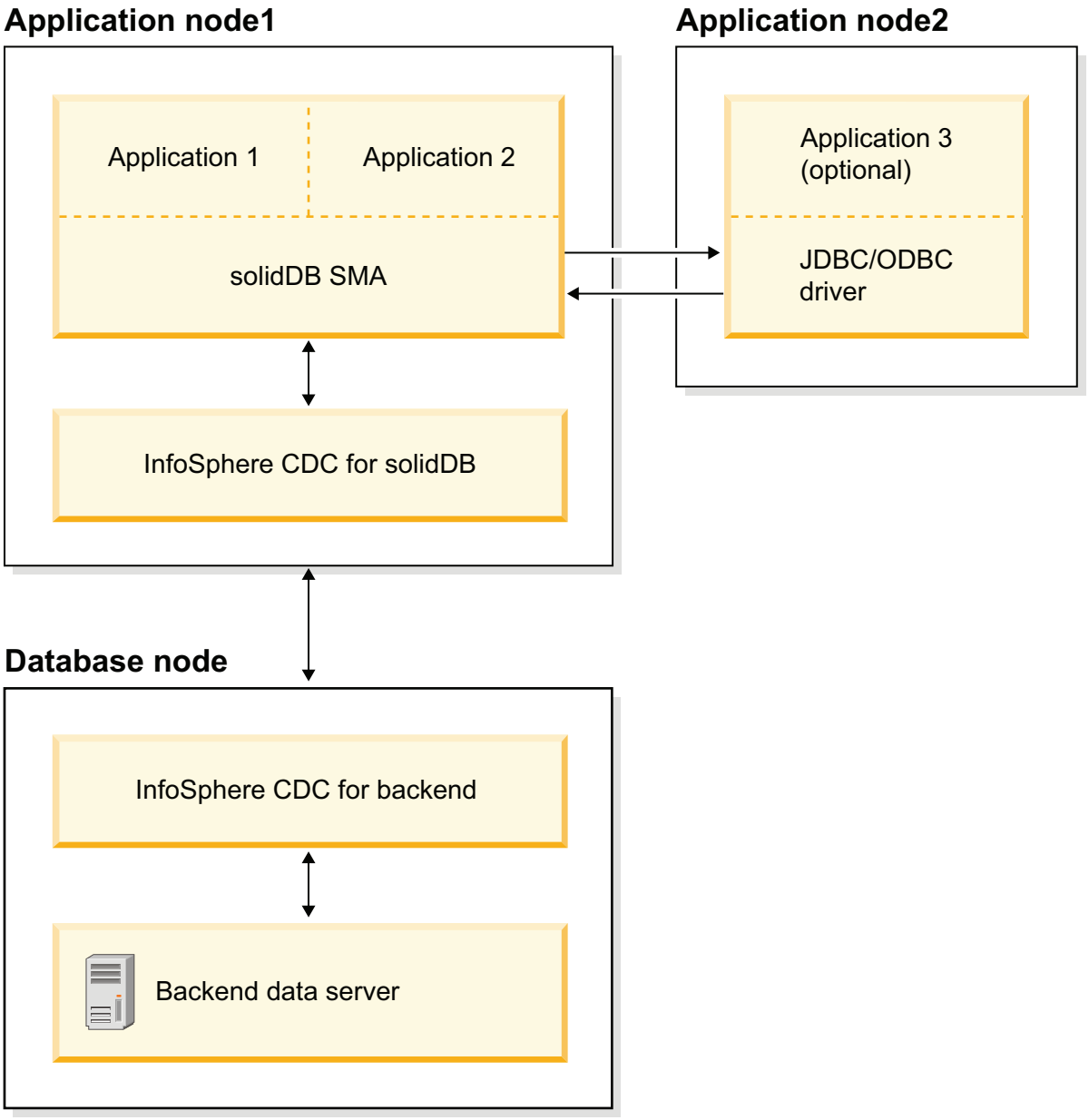

Figure 5. Universal Cache with shared memory access (SMA)

1.3.6 Universal Cache with High Availability

If the application requires uninterrupted access to data, the High Availability (HotStandby) feature insolidDB can be used to provide high availability. All HotStandby features, such as failovers in the solidDB server, multiple durability semantics, and read-on-standby are applicable to Universal Cache high availability setups. When you use HotStandby, the operation of Universal Cache is protected against single failures in the cache tier.

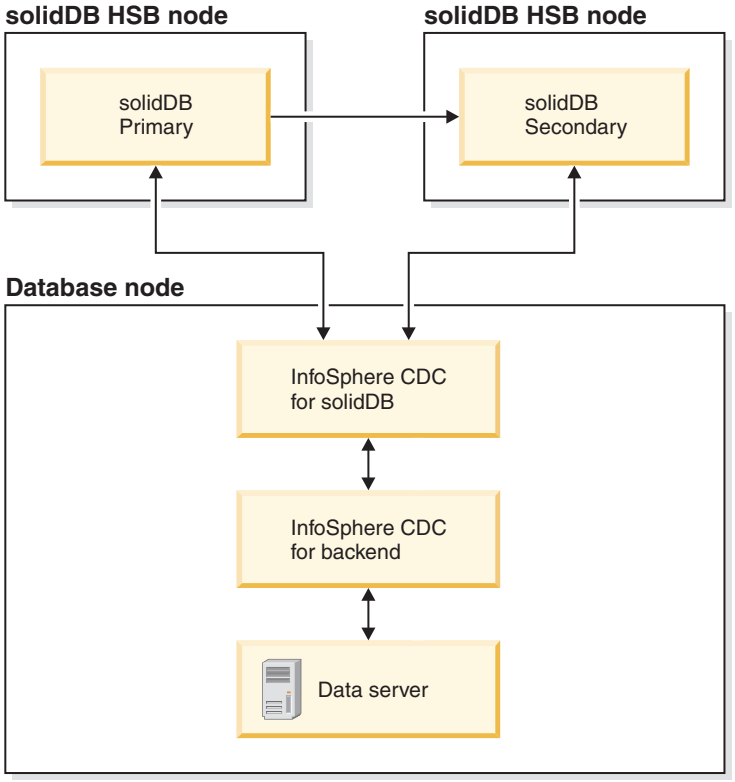

Figure 6. Universal Cache deployment with solidDB High Availability

Important: When designing deployments with HotStandby, remember to plan how failure scenarios are handled in your environment. See section Failure handling in Universal Cache for information about how failures are handled in the Universal Cache system.

Universal Cache deployment with backend high availability

Applications can have additional high availability or disaster recovery requirements on the backend data server as well. Different data servers implement such features using different technologies. The backend data server is still a single logical entity, but data and running processes can be distributed across multiple nodes.

In a typical backend high availability case, the InfoSphere CDC instances will run on the primary backend data server node. However, in case of the backend failure, the InfoSphere CDC instances will have to be restarted on a different machine as a part of the complete backend data server failover. The subscriptions will also have to be reconfigured to reconnect the relevant instances.

1.3.7 Application-driven data aging

The data aging feature enables applications to remove outdated data from the cache database while preserving it in the backend database. Data aging helps controlling main memory consumption in the cache; some of the data in the cache can become obsolete and it is not used by the applications. Removing the obsolete data frees main memory resources in the cache.

1.3.8 Security and authentication

- The cache and backend databases are protected by a traditional authentication mechanism in which a user has to provide a valid user ID and password combination to connect to a database. You can also use the operating-system-based external authentication mechanism available with solidDB.
- The user account that is used for the InfoSphere CDC instances to access the databases has to have sufficient privileges to access and modify the data and metadata. A database administrator username is recommended.
- The InfoSphere CDC instances uses scrambling to store the user names and passwords, that is, the authentication data is encrypted using a weak encryption method.
- The traffic between the InfoSphere CDC replication engines and the databases is encrypted in all cases. The encryption is provided, or available, in the database-specific JDBC drivers.
- The traffic between InfoSphere CDC replication engines is unencrypted.

1.3.9 Restrictions

Some restrictions apply to using the InfoSphere CDC technology to replicate data between solidDB and other data servers.

Restrictions on solidDB server in Universal Cache deployments

The following restrictions apply to solidDB server when used as a source or target data server in Universal Cache deployments.

Referential integrity

- **Referential integrity** (solidDB as a source and target)

  For continuous mirroring, referential integrity constraints (foreign keys) are allowed both on the source and target. The mandatory requirement is that the referential integrity associations are to be confined within the subscription; no foreign keys may point to tables outside the subscription. If this rule is violated, referential integrity errors can happen on the target during the mirroring, which ends the replication subscription.

  Referential integrity is not supported with automatic creation of tables.

- **Primary key constraints** (solidDB as a source)

  Primary keys are recommended but not mandatory. If no primary key is defined on a table, the execution of inserts and updates is less efficient than with primary keys. Primary key updates are restricted in the following way:

  – If the primary key is defined over a single column, no multi-row updates of the primary key are allowed.

  – In the case of multicolumn primary keys, multi-row updates are allowed if they affect a part of the primary key only.

  If any of the above rules are violated, an error is produced and replication on the subscription (mirroring) ends.

Datatype support restrictions

- LOB data types are not supported in D-tables (solidDB as a source)

  The large-size LOBs (maxiLOB, up to 2 GB) in D-tables (disk-based tables) are not supported in the source. If you attempt to write a maxiLOB into a D-table that is part of a log reader partition, the write fails and an error is returned to the application.

All LOBs maintained in M-tables (in-memory tables) that are within the available size limits (miniLOBS) are allowed. The size limit depends on the row size and the block size. With an assumption of a single LOB per row, the size limit is close to the block size. If the block size is set to 32 KB, a practical miniLOB size limit is about 30 KB.

- Limited LOB support (solidDB as a target)

  If a LOB is written into an M-table and exceeds the miniLOB size limit, an error is returned and replication on the subscription ends.

TRUNCATE

In subscriptions where solidDB is a source, the TRUNCATE TABLE statement is not allowed on tables which are part of a subscription. If this rule is violated, an error is returned to the application.

Transient and temporary tables

Because non-persistent (transient and temporary) tables are not logged, if solidDB is a source datastore, transient and temporary tables cannot be part of a subscription. Transient and temporary tables can be used in a subscription where solidDB is the target datastore.

Multiple NULLs in UNIQUE columns

In subscriptions where solidDB is a target, there can be at most one NULL instance in a column defined as UNIQUE. An effort to propagate insertion of an additional NULL results in the UNIQUE constraint violation and replication on the subscription (mirroring) ends.

Data and workload partitioning using multiple cache databases

Multiple solidDB servers can be used for data and workload partitioning; the backend data can be distributed (partitioned) over several cache databases. However, each cache database is autonomous and processes the application requests only within the partition it holds, without accessing data in any of the other cache databases (partitions).

Referential integrity constraints apply also; the partition cannot contain tables which are referencing or referenced outside the partition.

DDL changes for tables included in InfoSphere CDC subscriptions

solidDB does not restrict renaming of tables that are part of an InfoSphere CDC subscription△△ △.

If you rename a table that is part of a subscription, the solidDB server might shut down unexpectedly, for example, when the InfoSphere CDC for solidDB instance is restarted.△△ If you need to rename or make other DDL changes to a table that is part of the InfoSphere CDC replication schema, follow the instructions in △△Updating, removing, and viewing tables for replication△△ in the △△*InfoSphere CDC Management Console Administration Guide*.

Restrictions on InfoSphere CDC in Universal Cache deployments

The following features that may be available in the InfoSphere CDC components for other data servers are not supported in InfoSphere CDC for solidDB.

- **Fast load for refresh**

  InfoSphere CDC for solidDB does not support the fast load for refresh feature.
- **Automatic creation of target tables**

  If the tables meant to be mirrored are associated with referential integrity constraints, you cannot use the option of creating target tables automatically (**Create new target tables**), while defining a new subscription. Instead, use the option **Map to existing tables**. If this rule is violated, the subscription will not be created.

  This restriction applies to all configurations, also with other DBMS products.
- **Row filtering**

  Row filtering (horizontal partitioning) is fully functional only if primary keys are defined on the source tables.
- **Dropping and re-creating tables when solidDB is the source datastore**

  If you need to drop and re-create tables in subscriptions where solidDB is the source datastore, you need to reconfigure the table mappings.

Part 2. Getting started with Universal Cache

2 Introduction to Universal Cache

The Universal Cache capability in IBM solidDB is a solution for speeding up traditional disk-based databases. It incorporates a high-speed solidDB in-memory database that caches performance-critical data from a disk-based database. The in-memory cache allows the processing of the application load by a fraction of the response time it takes to be executed in the backend database. This improves performance, speed, and latency.

Architecture

The architecture of Universal Cache is based on three main components: the solidDB (in-memory) database (*cache*), a relational database server (*backend*), and the InfoSphere CDC data synchronization software that copies data to and from the cache and the backend.

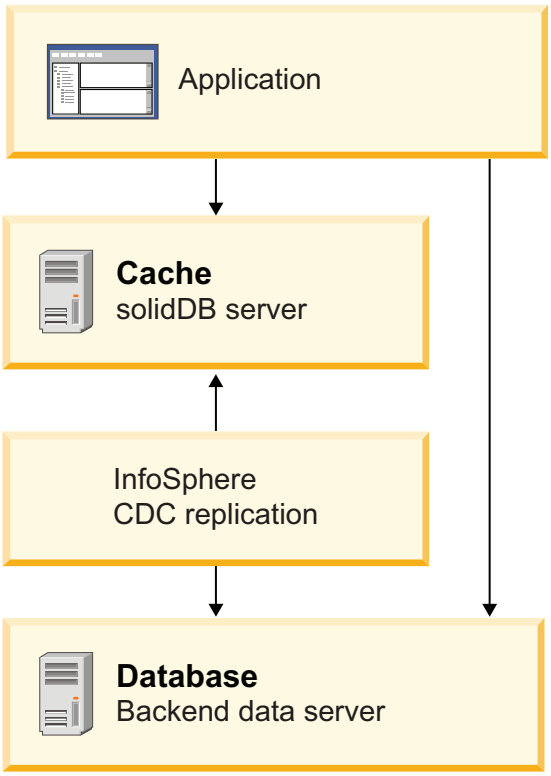

Figure 7. Architecture of solidDB Universal Cache

Principles of operation

To use Universal Cache, you must first identify the data you want to cache and configure the environment accordingly. The data can then be loaded from the backend database to the cache, so that when applications run against the cache database, they can take advantage of high performance and low latency of the solidDB server. As changes are made to the data, the InfoSphere CDC replication technology synchronizes data between the cache database and the backend database. Additionally, the SQL passthrough functionality enables applications to access data in both databases with a single interface.

Tools and administration

The Universal Cache includes both graphical and command-line tools and utilities for installing, configuring, and administering the Universal Cache environment.

- The **InfoSphere CDC Management Console** is an interactive application with a GUI that you can use to configure and monitor replication of data.
- **`dminstancemanager`** and **`dmsubcriptionmanager`** are command-line utilities that make it possible to script InfoSphere CDC instance and subscription management.

2.1 Installation topologies for Universal Cache

You can install the Universal Cache components on the same server for a simple evaluation topology, or on independent servers for a production-level topology.

General principles

- There can be several solidDB cache databases in the Universal Cache deployment, but only one backend data server.
- Typically, the InfoSphere CDC instance is created on each node participating in InfoSphere CDC replication.
- The solidDB server and the InfoSphere CDC for solidDB instance do not need to be located on the same node.

  This is because InfoSphere CDC for solidDB can read and insert data into a solidDB database using both local and remote JDBC connection.
- In configurations using solidDB High Availability (HotStandby), the InfoSphere CDC instance must run on a different node than the solidDB server.

2.1.1 Example: Evaluation topology

In a typical evaluation setup, all the Universal Cache components are installed on a single computer, except for the backend dataserver. Typically you would also already have a working installation of the backend data server that contains the data you want to cache into an in-memory solidDB database.

The evaluation topology is appropriate for exploring the software or for training environments, but not for production environments.

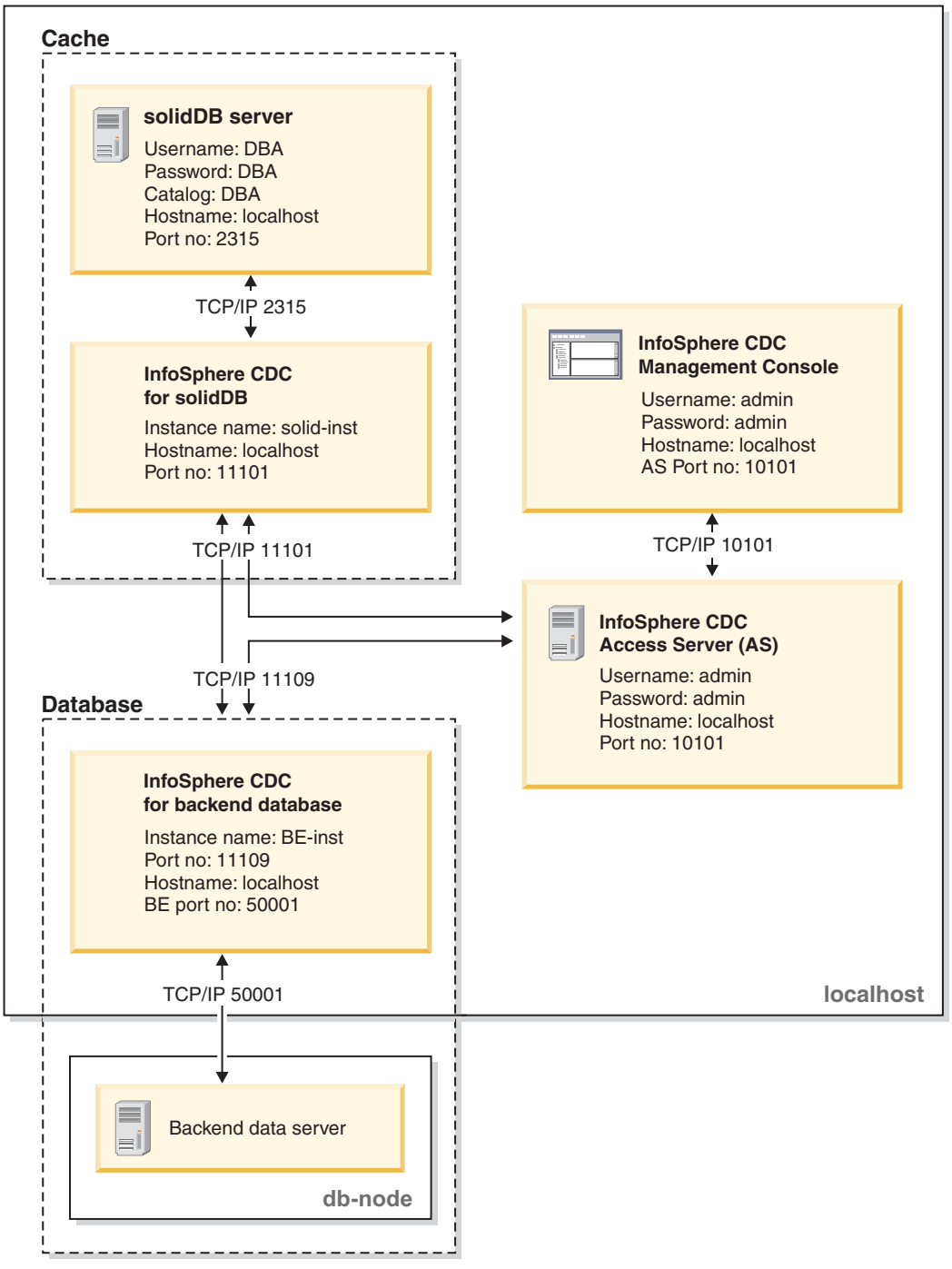

Figure 8. Universal Cache - evaluation topology

2.1.2 Example: Production topology

In a typical production setup, the cache and database components are installed on separate server machines and tooling is located on a management node. Access Server can be located, for example, on the backend database node.

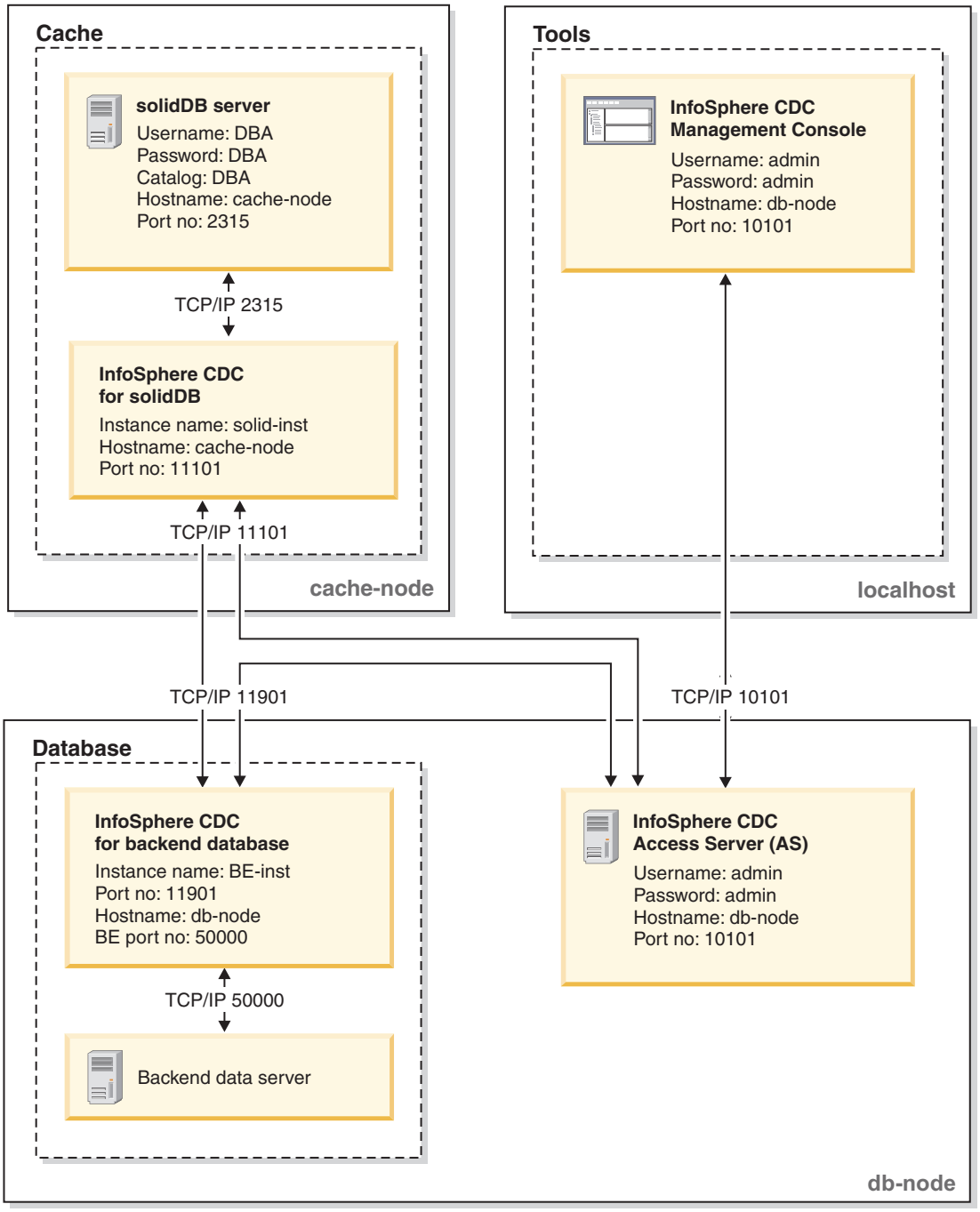

Figure 9. Typical Universal Cache deployment topology - production

2.1.3 Example: Multiple cache databases topology

Multiple solidDB servers can be used, for example, for partitioning backend data over several solidDB cache databases.

Note: In a deployment with multiple cache databases, each solidDB server is autonomous and processes the application requests without accessing data in any of the other solidDB servers.

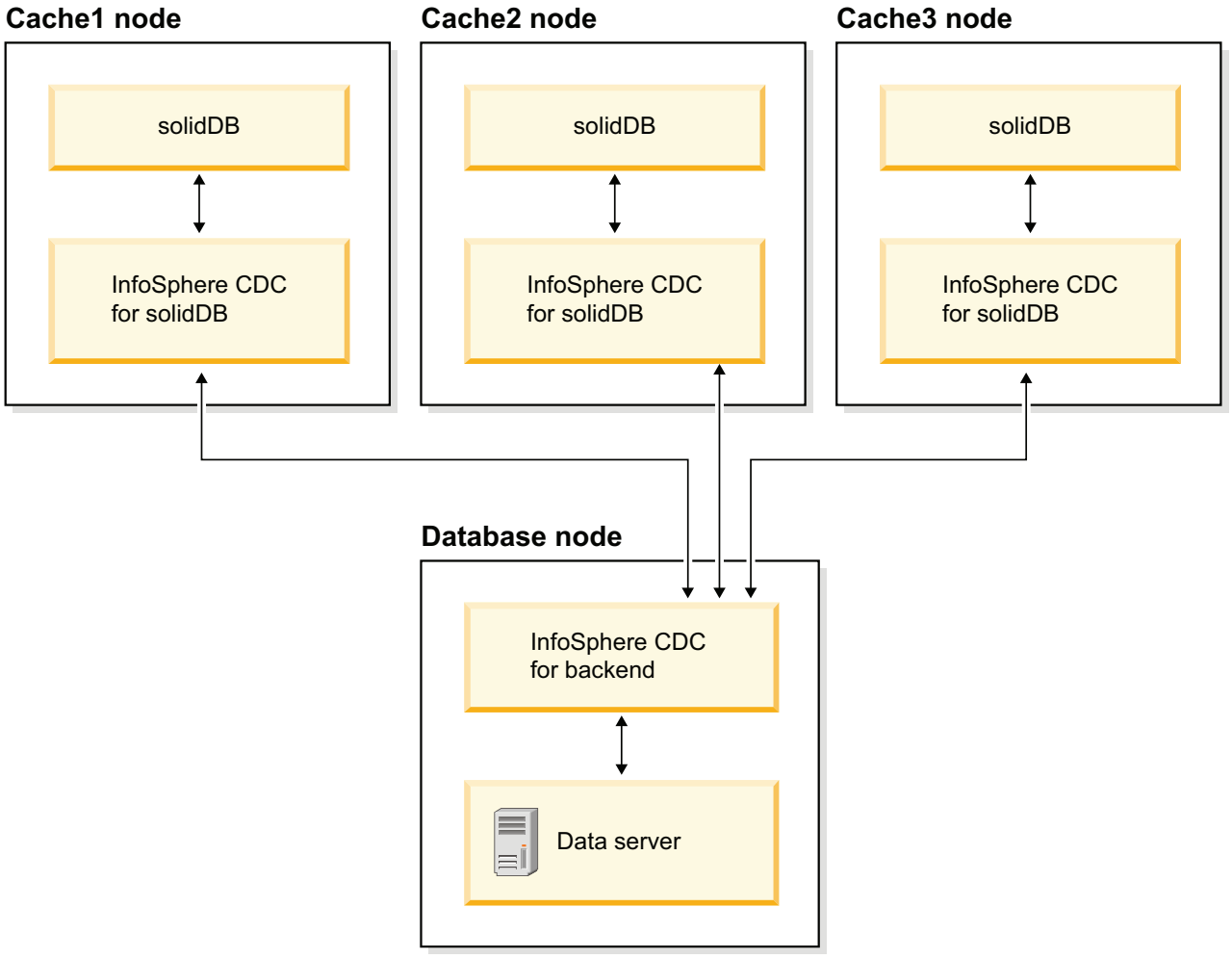

Figure 10. Universal Cache deployment with multiple solidDB servers

Note: The above topology illustration does not include the Access Server or the management tools. The Access Server would typically be located on the Database node and the management tools on a separate management node.

2.1.4 Example: Universal Cache with High Availability topology

In a typical HotStandby setup, all InfoSphere CDC instances will run on the backend database node, and the connection to the HotStandby pair is established remotely. The management tools run on a separate node.

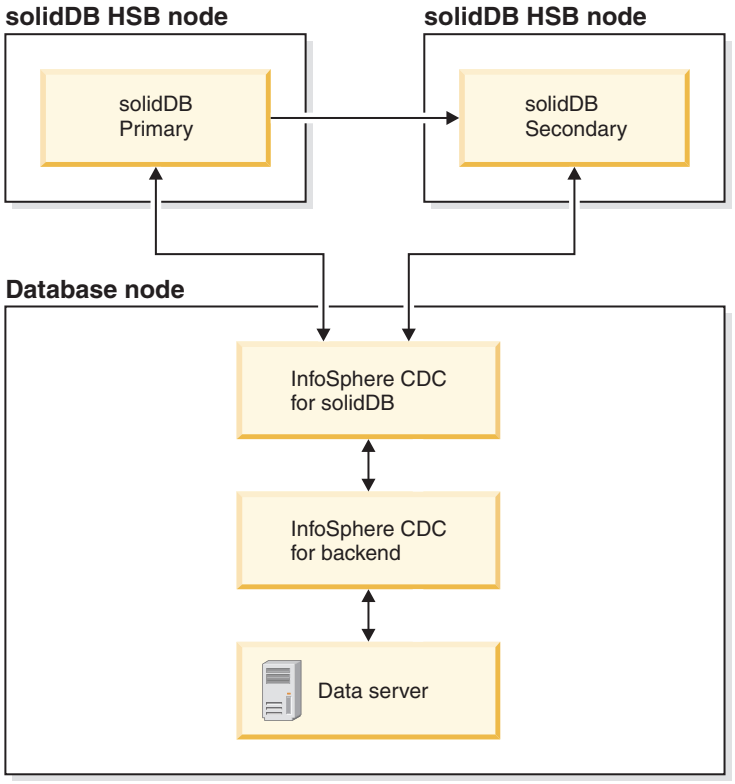

Figure 11. Example: Universal Cache with solidDB High Availability

Note: The above topology illustration does not include the Access Server or the management tools. The Access Server would typically be located on the Database node and the management tools on a separate management node.

2.2 Component and installation package information

The Universal Cache setups include both solidDB and InfoSphere CDC components. To install Universal Cache, you need the installation packages shown in the Universal Cache column in the table below. You need to install each Universal Cache component separately.

Table 3. solidDB V7.0 installation packages

Component	solidDB	solidDB with InfoSphere CDC replication	solidDB with Universal Cache
IBM solidDB 7.0	X	X	
IBM InfoSphere Change Data Capture solidDB 7.0		X	
IBM InfoSphere Change Data Capture Access Server 6.5		X	
IBM InfoSphere Change Data Capture Management Console 6.5		X	

Table 3. solidDB V7.0 installation packages (continued)

Component	solidDB	solidDB with InfoSphere CDC replication	solidDB with Universal Cache
IBM InfoSphere Change Data Capture *backend data server* 6.5 One of the following: • IBM InfoSphere Change Data Capture DB2 Linux, UNIX, and Windows 6.5 • IBM InfoSphere Change Data Capture Informix 6.5 • IBM InfoSphere Change Data Capture Microsoft SQL Server 6.5 • IBM InfoSphere Change Data Capture Oracle Trigger 6.5 • IBM InfoSphere Change Data Capture Oracle Redo 6.5 • IBM InfoSphere Change Data Capture Sybase 6.5 • IBM InfoSphere Change Data Capture DB2 z/OS® 6.5 • IBM InfoSphere Change Data Capture DB2 iSeries® 6.1			X
IBM Data Server Driver for ODBC and CLI 9.7 **Note:** Needed only in Universal Cache configurations with SQL passthrough when the backend data server is an IBM data server.			X
IBM solidDB 7.0 License Certificate	X	X	X
IBM solidDB 7.0 Documentation	X	X	X
IBM InfoSphere Change Data Capture Documentation 6.5		X	X

2.2.1 solidDB server package

The solidDB server package contains a complete set of the server software, including the JDBC and ODBC drivers and various utility programs.

The solidDB server package is delivered with an evaluation license certificate file, `solideval.lic`. With the evaluation license, you can evaluate solidDB for 90 days. For acquiring a permanent license, contact IBM Corporation.

Table 4. solidDB installation images

Component name	Installation package
IBM solidDB 7.0	**Linux and UNIX** solidDB-7.0-<platform>.bin **Windows** solidDB-7.0-<platform>.exe

Directory structure

The default installation of solidDB 7.0 creates a directory called `solidDB7.0`.

The files and subdirectories in the `solidDB7.0` installation directory are explained in the following table.

Table 5. `solidDB7.0` directory structure

Location	Explanation
Root directory	The root directory contains, for example: • a script that is used to facilitate running samples in the database evaluation phase • the evaluation license file • the `welcome.html` file for accessing the package documentation
`bin`	solidDB binary files and dynamic library files

`bin/C`
 	
`bin/N`
	Auxiliary libraries for IBM Global Security Kit (GSKit)

`doc_html`, `doc_txt`	Package documentation in HTML and text format
`eval_kit/standalone`	Working directory for an evaluation version of the solidDB server. This directory contains a sample `solid.ini` configuration file and an evaluation license file (`solideval.lic`).
`eval_kit/cdc`	Working directory for an evaluation version of the solidDB server for use with Universal Cache or InfoSphere CDC replication. This directory contains a sample `solid.ini` configuration file and an evaluation license file (`solideval.lic`).
`include`	C program headers
`jdbc`	solidDB JDBC Driver Data store helper archive for use with WebSphere® (`SolidDataStoreHelper.jar`) solidDB dialect for Hibernate (`SolidSQLDialect.jar`)
`lib`	Static linkable library files
`lib32`	32-bit static linkable library files – 64-bit AIX® and Solaris packages only The 32-bit libraries can be installed on 64-bit systems. The 64-bit libraries cannot be installed on 32-bit systems.
`licence`	License and notices files
`manuals`	The English versions of the manuals in PDF format can be downloaded to this folder and then accessed through the **Manuals** link on the **Welcome** page
`procedures`	SQL scripts for creating and running stored procedures for data aging and refresh
`properties`	Metadata for IBM Tivoli® Usage and Accounting Manager
`samples`	Samples that can be used in the database evaluation phase and future application development

Library file names

The solidDB server provides many files as linkable libraries.

Most of the library files fall into one of the following categories:

- ODBC drivers
- Shared memory access and linked library access files
- Communication library files
- SA (Server API) library file

All platforms do not have every file. For example, some communication library files are available on Windows environments only.

Some library files are static, that is, they are linked to the client application executable program when you do a compile-and-link operation. Other library files are dynamic: these files are stored separately from your executable program and are loaded into memory when your program runs. For many libraries, the solidDB server provides both a static and a dynamic version on some or all platforms.

Library files are found in the following two directories:

- `bin`
- `lib`

As a rule, the `bin` directory contains dynamic libraries (in addition to executable files), while the `lib` directory contains static libraries. On Windows environments, the `lib` directory also contains the import libraries.

Additionally, on Windows environments, the ODBC and communication `.dll` libraries are copied to the `C:\Windows\system32` directory.

If you use the 32-bit installation program to install the solidDB server on a 64-bit environment, the `.dll` library files are copied to the `C:\Windows\SysWOW64` directory.

The exact library file names depend on the platform. See the following tables for examples on Windows and Linux environments:

Table 6. Example: solidDB library files in Windows 32-bit package

File name	Description
bin\	
sacw3270.dll	ODBC library - ASCII
snpw3270.dll	NamedPipes communication protocol link library
socw3270.dll	ODBC library - Unicode
sosw3270.dll	ODBC Driver Manager setup library
ssaw3270.dll	solidDB SA API library
ssolidac70.dll	Linked library access (LLA) dynamic library
stcw3270.dll	TCP/IP communication protocol link library
lib\	

Table 6. Example: solidDB library files in Windows 32-bit package (continued)

File name	Description
solidctrlstub.lib	solidDB Control API (SSC) stub library.
	This static library is used if you want to write code that can be run either locally with the linked library access library, or remotely without the linked library access.
solidimpac.lib	Linked library access (LLA) import library
solidimpodbca.lib	ODBC import library - ASCII
solidimpodbcu.lib	ODBC import library - Unicode
solidimpsa.lib	solidDB SA API import library

Table 7. Example: solidDB library files in Linux 32-bit package

File name	Description
bin\	
sacl2x70.so	ODBC shared library - ASCII
socl2x70.so	ODBC shared library - Unicode
ssal2x70.so	solidDB SA API library
ssolidac70.so	Linked library access (LLA) shared library
ssolidsma70.so	Shared memory access (SMA) shared library
lib\	
solidctrlstub.a	solidDB Control API (SSC) stub library.
	This static library is used if you want to write code that can be run either locally with the linked library access library, or remotely without the linked library access.
solidac.a	Linked library access (LLA) static library
solidodbca.a	ODBC static library - ASCII
solidodbcu.a	ODBC static library - Unicode
solidsa.a	solidDB SA API static library
libssolidac70.so	Symbolic link for shared LLA library
libssolidsma70.so	Symbolic link for shared SMA library
libsacl2x70.so	Symbolic link for shared ODBC library - ASCII
libsocl2x70.so	Symbolic link for shared ODBC library - Unicode
libssal2x70.so	Symbolic link for shared solidDB SA API library
libsolidodbca.a	Symbolic link for static ODBC library - ASCII
libsolidodbcu.a	Symbolic link for static ODBC library - Unicode
libsolidsa.a	Symbolic link for static solidDB SA API library
libsolidac.a	Symbolic link for static LLA library

For a list of the library file names on your installation of solidDB server, see the SDK Notes in the solidDB package, accessible through the **Welcome** page in your solidDB installation directory.

Dynamic library file naming conventions

Dynamic library files use the following naming convention:

```
sLLpppVV.eee
```

where

- `LL` = purpose of the library
  - `ac`: ODBC library - ASCII
  - `np`: NamedPipes communication protocol link library
  - `oc`: ODBC library - Unicode
  - `os`: ODBC Driver Manager setup (for Windows only)
  - `sa`: solidDB SA API library
  - `solidac`: Linked library access (LLA) dynamic library
  - `solidsma`: Shared memory access (SMA) dynamic library
  - `tc`: TCP/IP communication protocol link library
- `ppp` = platform
  - `a5x64`: AIX, 64-bit
  - `hia64`: HP-UX 11 64-bit (IA64)
  - `l2x`: Linux for x86
  - `l2x64`: Linux for x86, 64-bit
  - `lzx64`: Linux for System z, 64-bit
  - `s0x64`: Solaris 10 (SPARC, 64-bit)
  - `s0xi64`: Solaris 10 (ix86, 64-bit)
  - `w32`: Windows 32-bit (x86)
  - `w64`: Windows 64-bit (x86)
- `VV` = first two digits of the solidDB version, for example 70 for version 7.0, 63 for version 6.3
- `eee` = platform-specific file name extension:
  - `*.dll` Dynamic Link Library for Windows
  - `*.so` (Shared Object) for AIX, HP-UX, Linux, and Solaris

ODBC, JDBC, and proprietary programming interfaces

The solidDB server provides ODBC, JDBC and proprietary interfaces for clients.

For more details, see the *IBM solidDB Programmer Guide*.

solidDB JDBC Driver 2.0

Table 8. solidDB JDBC Driver 2.0 key information

Compatibility	JDBC 2.0, with selected features of JDBC 2.0 Optional Package
Driver location	`<solidDB installation directory>/jdbc/SolidDriver2.0.jar`

JDBC URL format	`jdbc:solid://<hostname>:<port>/<username>/<password>[?<property-name>=<value>]...`
	For example:
	`"jdbc:solid://localhost:1964/dba/dba"`

Driver class name	solid.jdbc.SolidDriver

Standard compliance

The solidDB JDBC 2.0 Driver supports the JDBC 2.0 specification. Additionally, Connection Pooling, JNDI Data Sources, and Rowsets of the JDBC 2.0 Optional Package (known before as Standard Extension) are supported too.

Non-standard features include support for IBM WebSphere and timeout control extensions.

The following features of the Optional Package are currently supported by the solidDB JDBC 2.0 driver:

- Connection pooling (class solid.jdbc.ConnectionPoolDataSource)
- Connected RowSet (class solid.jdbc.rowset.SolidJDBCRowSet)
- Implemented JDBC data sources:
  - solid.jdbc.DataSource (implements javax.sqlDataSource)
  - solid.jdbc.SolidConnectionPoolDataSource (implements javax.sql.ConnectionPoolDataSource)
- JTA (Java Transaction API) – XA interface for Java™ (implements javax.transaction.xa.XAResource and javax.transaction.xa.Xid)

Full documentation for the solidDB JDBC Driver is included in the *IBM solidDB Programmer Guide*.

solidDB JDBC Driver extensions

The solidDB JDBC Driver supports the following non-standard extensions. For more information, see the *IBM solidDB Programmer Guide*.

JDBC URL format

You can set the connection property values in the URL string.

Connection timeout

Connection timeout refers to the response timeout of any JDBC call that invokes data transmission over a connection socket. If the response message is not received within the time that is specified, an I/O exception is thrown. The JDBC standard (2.0/3.0) does not support setting of the connection timeout. The solidDB product has two ways for doing that: one using a non-standard driver manager extension method and the other using the property mechanisms. The time unit in either case is 1 ms.

Login timeout

The timeout fires at the connect time. The setting is implemented with a connection property. The property overrides the login timeout for JDBC specified by other means (like login timeout parameter in Driver Manager).

Connection idle timeout

If the connection is inactive for the amount of time specified with the idle timeout property, the server closes the connection. The connection idle timeout property overrides the server parameter setting for the session.

Statement cache

You can set the size of the statement cache for a connection.

Transparent Connectivity Support

solidDB JDBC driver fully supports solidDB Transparent Connectivity (TC) including transparent failover and load balancing. See the *IBM solidDB High Availability User Guide* for more information about usage of Transparent Connectivity.

Shared memory access (SMA) connection property
The SMA connection property defines that the driver connects to a SMA server with a local connection, bypassing network protocols.

SQL passthrough connection properties
The SQL passthrough connection property defines the default passthrough mode for the connection.

Catalog and schema name connection properties
You can set the catalog and schema names for the connection.

WebSphere support
To support WebSphere, adata source adapter SolidDataStoreHelper is provided in a separate file `SolidDataStoreHelper.jar`, in the `'jdbc'` directory of thesolidDB package.

solidDB ODBC Driver 3.5.x

solidDB provides two ODBC drivers, one for Unicode and one for ASCII character sets. For more information about these drivers, see the *IBM solidDB Programmer Guide*.

The following functions are not supported:

- SQLBrowseConnect
- SQLSetScrollOptions
- SQLParamOptions
- SQLNativeSql
- SQLMoreResults

ODBC extensions
solidDB ODBC driver incorporates several extensions for, for example, timeout controls, statement cache behavior, and support for Transparent Connectivity. For more information, see the *IBM solidDB Programmer Guide*.

Proprietary interfaces

The solidDB Application Programming Interface (SA API) and solidDB Server Control API (SSC API) allow, for example, C programs to directly call functions inside the database server. These proprietary interfaces are provided with the solidDB shared memory access (SMA) and linked library access (LLA) libraries.

System tools and utilities

The solidDB server package includes console tools for data management and administration, and command-line utilities for data export and import.

The tools and utilities are available in the `'bin'` directory in the solidDB server installation directory.

Console tools

solidDB SQL Editor (`solsql`)
solidDB SQL Editor (**`solsql`**) is a console tool that you can use to issue SQL statements and solidDB ADMIN COMMANDs at the command prompt. You can also execute script files that contain the SQL statements.

solidDB Remote Control (`solcon`)
solidDB Remote Control (**`solcon`**) is a console tool for administration; users with administrator rights can issue ADMIN COMMANDs at the command

prompt or by executing a script file that contains the commands. With **solcon**, the ADMIN COMMANDs can be issued as part of the **solcon** startup command line.

Because only users with administrator rights can access **solcon**, if only **solcon** is deployed at a production site, the administrators cannot accidentally execute SQL statements that could change the data.

Tools for exporting and loading data

solidDB Speed Loader (solloado or solload)
solidDB Speed Loader (**solloado** or **solload**) loads data from an external file into a database.

solidDB Export (solexp)
solidDB Export (**solexp**) exports data from a database into a file. It also creates control files used by solidDB Speed Loader (**solloado** or **solload**) to perform data load operations.

solidDB Data Dictionary (soldd)
solidDB Data Dictionary (**soldd**) exports the data dictionary of a database. It produces an SQL script that contains data definition statements that describe the structure of the database.

2.2.2 InfoSphere CDC packages

The InfoSphere CDC components are delivered as separately deployable packages.

InfoSphere CDC for solidDB

The InfoSphere CDC for solidDB package contains the software for the replication engine that captures and transfers data changes between solidDB and other databases.

Table 9. InfoSphere CDC for solidDB installation images

Component name	Installation package	Contents

InfoSphere Change Data Capture solidDB	**Linux and UNIX**
	`setup-cdc-<platform>-solid.bin`
	For example:
	`setup-cdc-linux-x86-solid.bin`
	Windows
	`setup-cdc-x86-solid.exe`
		• Software for the configuration tool and the InfoSphere CDC instance for solidDB
		• solidDB JDBC Driver (`SolidDriver2.0.jar` in the `/lib` directory)
		• Tools, utilities, and samples (`/samples` directory)
		– Automation tools, utilities, and samples for scripting most common InfoSphere CDC tasks (`ucutils`, `ucpassthrough`, and `uchsbmonitor` directories)
		– Generic InfoSphere CDC samples for Java user exits and SQL scripts
		• InfoSphere CDC API documentation (`/docs` directory)

InfoSphere CDC for backend

The InfoSphere CDC for backend package contains the software for the replication engine that captures and transfers data changes between the backend and solidDB databases.

Table 10. InfoSphere CDC for backend installation images

Component name	Installation package	Contents
InfoSphere Change Data Capture *for a backend data server* • IBM InfoSphere Change Data Capture DB2 Linux, UNIX, and Windows 6.5 • IBM InfoSphere Change Data Capture Informix 6.5 • IBM InfoSphere Change Data Capture Microsoft SQL Server 6.5 • IBM InfoSphere Change Data Capture Oracle Trigger 6.5 • IBM InfoSphere Change Data Capture Oracle Redo 6.5 • IBM InfoSphere Change Data Capture Sybase 6.5 • IBM InfoSphere Change Data Capture DB2 z/OS 6.5 • IBM InfoSphere Change Data Capture DB2 iSeries 6.1	**Linux and UNIX:** setup-<platform>-<backend_dataserver>.bin For example: setup-aix-power-udb.bin **Windows:** setup-x86-<backend_dataserver>.exe	• Software for the configuration tool and the InfoSphere CDC instance for *backend data server* • PDF format *InfoSphere Change Data Capture, End-User Documentation* (`/docs` directory) • Sample Java user exits and SQL scripts (`/samples` directory) • InfoSphere CDC API documentation (`/docs` directory)

InfoSphere CDC Access Server

The InfoSphere CDC Access Server package contains the software for controlling access to the replication environment.

Table 11. InfoSphere CDC for Access Server installation images

Component name	Installation package	Contents

InfoSphere Change Data Capture Access Server	**Linux and UNIX**
	`cdcaccess-<version>-setup.bin`
	For example:
	`cdcaccess-6.5.1618.0-solaris-sparc-setup.bin`
	Windows
	`cdcaccess-<version>-setup.exe`
	For example:
	`cdcaccess-6.5.1618.0-setup.exe`
		• Software for controlling access to your replication environment

InfoSphere CDC Management Console

The InfoSphere CDC Management Console package contains the software for configuring and monitoring user access and replication subscriptions. Management Console is available only on Windows environments.

Table 12. InfoSphere CDC Management Console installation images

Component name	Installation package	Contents
InfoSphere Change Data Capture Management Console	**Linux and UNIX** Not applicable, Management Console is available only on Windows environments **Windows** `cdcmc-<version>-setup.exe` For example:`cdcmc-6.5.1618.0-setup.exe`	• Software for configuring and monitoring InfoSphere CDC user access and replication subscriptions • PDF format *InfoSphere Change Data Capture Management Console, Administration Guide* (`/documentation` directory) • Online help (accessible through **Help** menu in the Management Console user interface) • IBM Java SDK and Runtime Environment Guides (`/docs` directory)

2.2.3 IBM Data Server Driver for ODBC and CLI package

The IBM Data Server Driver for ODBC and CLI is delivered as a compressed file. It is used with the SQL passthrough feature in Universal Cache if the backend data server is an IBM data server.

- Windows operating systems:
  `ibm_data_server_driver_for_odbc_cli_<platform>.zip`
- Linux and UNIX operating systems:
  `ibm_data_server_driver_for_odbc_cli_<platform>.tar.Z`

There is no installation program for the IBM Data Server Driver for ODBC and CLI. Instead, you must install the driver manually by uncompressing the file.

2.2.4 Documentation packages

Documentation for solidDB is composed of the *IBM solidDB Documentation* package and the *InfoSphere Change Data Capture Documentation* package. Both packages are available as an online information center and in PDF format.

solidDB documentation

solidDB documentation is available online in the solidDB 7.0 Information Center and in PDF format. Most up-to-date information is always available in the Information Center.

Delivery of solidDB documentation

solidDB 7.0 Information Center

The most up-to-date solidDB documentation is available in the information center format at http://publib.boulder.ibm.com/infocenter/soliddb/v7r0/.

solidDB manuals in PDF format

The PDF manuals are available for download at the following locations:

- Software Support portal for solidDB at ftp://ftp.software.ibm.com/software/data/soliddb/info/7.0/man/.
- IBM Publications Center at http://www.elink.ibmlink.ibm.com/publications/servlet/pbi.wss

In addition, the PDF format manuals are available as the *IBM solidDB Documentation* package. This package is delivered together with the software packages in IBM Passport Advantage®, or in the Quick Start DVD in physical media deliveries.

Tip: If you download the English version PDF files to the `manuals` directory in your solidDB server installation directory, you can access the manuals also through the **Welcome** page of your solidDB software package. For detailed instructions, see section Installing solidDB Documentation package.

InfoSphere CDC documentation

InfoSphere CDC for solidDB documentation is included in the *IBM solidDB Documentation* package. Documentation for InfoSphere CDC Management Console, InfoSphere CDC Access Server, and InfoSphere CDC engine for the back-end data server is part of the *InfoSphere Change Data Capture Documentation* package.

Delivery and location of documentation for InfoSphere CDC components

The *InfoSphere Change Data Capture Documentation* package is available in information center and PDF format:

- IBM InfoSphere Change Data Capture version 6.5 Information Center
- InfoSphere Change Data Capture 6.5 End-User Documentation in PDF format - IBM Software Support Portal
- Embedded Help accessible through the Management Console **Help** menu
- *InfoSphere Change Data Capture Documentation* installation package (PDF format), available at Passport Advantage

2.3 System requirements for Universal Cache

The solidDB product family supports more than 30 different platforms, each understood as a combination of hardware type and operating system. Typically all commonly used platforms are supported. Support for legacy platforms might be available upon request.

2.3.1 IBM solidDB supported platforms

The following table shows an overview of the supported platforms for the components included in the IBM solidDB 7.0 product offering.

More detailed information about the platform support for each component is available through the Software product compatibility reports portal on ibm.com® (see direct links after the table).

Table 13. IBM solidDB supported platforms

Operating system		Hardware	solidDB server 7.0	InfoSphere CDC 6.5											ODBC
				InfoSphere CDC solidDB 7.0	MC	AS	DB2	DB2 z/OS	DB2 iSeries	IDS	OR	OT	MS SQL	Sybase	
AIX	AIX 7.1 AIX 6.1	64-bit systems with POWER5, POWER6®, or POWER7®	X	X		X	X			X	X			X	X
HP-UX	HP-UX 11i v3	Itanium-based HP Integrity Series systems	X	X		X				X	X			X	X

Table 13. IBM solidDB supported platforms (continued)

Operating system		Hardware	solidDB server 7.0	InfoSphere CDC 6.5											ODBC
Linux	Red Hat Enterprise Linux (RHEL) 6, 5 SUSE Linux Enterprise Server (SLES) 11, 10	32-bit and 64-bit systems based on Intel or AMD processors that are capable of running the supported Linux operating systems (x86 and x64 systems)	X	X		X	X			X	X			X	X
	Red Hat Enterprise Linux (RHEL) 5 System z SUSE Linux Enterprise Server (SLES) 10 System z	System z	X[1]				X					X			X
Solaris	Solaris 11	64-bit systems with UltraSPARC processors	X[3]												
		64-bit systems with x86 processors	X[3]												
	Solaris 10	64-bit systems with UltraSPARC processors	X	X		X	X			X	X			X	X
		64-bit systems with x86 processors	X												X
Windows	Windows Server 2012 (Standard Server, Enterprise Server, and Datacenter Editions) Windows 8 (Professional, Enterprise, and Ultimate editions)	32-bit and 64-bit systems based on Intel or AMD processors that are capable of running the supported Windows operating systems (x86 and x64 systems)	X[2]												
	Windows Server 2008 R2, 2008 (Standard Server, Enterprise Server, and Datacenter Editions) Windows 7 (Professional, Enterprise, and Ultimate editions) Windows Vista (Business, Enterprise, and Ultimate editions)		X	X	X	X	X			X			X	X	X
IBM i	i5/OS™ 7.1 i5/OS 6.1 i5/OS 5.4 i5/OS 5.3	POWER® System with i5 processors							X						X
z/OS	z/OS V1.11 z/OS V1.10	System z						X							X

Table 13. IBM solidDB supported platforms (continued)

Operating system	Hardware	solidDB server 7.0	InfoSphere CDC 6.5	ODBC
MC = InfoSphere Change Data Capture Management Console 6.5				
AS = InfoSphere Change Data Capture Access Server 6.5				
DB2 = InfoSphere Change Data Capture DB2 Linux, UNIX, and Windows 6.5				
DB2 z/OS = InfoSphere Change Data Capture DB2 z/OS 6.5				
DB2 iSeries = InfoSphere Change Data Capture DB2 iSeries 6.1				
IDS = InfoSphere Change Data Capture Informix 6.5				
OR = InfoSphere Change Data Capture Oracle Redo 6.5				
OT = InfoSphere Change Data Capture Oracle Trigger 6.5				
MS SQL = InfoSphere Change Data Capture Microsoft SQL Server 6.5				
Sybase = InfoSphere Change Data Capture Sybase 6.5				
ODBC = IBM Data Server Driver for ODBC and CLI 9.7				
[1] Support for System z introduced in V7.0 Fix Pack 1				
[2] Support for Windows 8 and Windows Server 2012 introduced in V7.0 Fix Pack 4				
[3] Support for Solaris 11 introduced in V7.0 Fix Pack 6				

Software product compatibility reports on ibm.com

The Software product compatibility reports portal on ibm.com provides various tools for generating reports on the hardware and software support level of IBM products. Use the following links to view reports specific to IBM solidDB 7.0.

- Operating systems for IBM solidDB 7.0
- IBM solidDB 7.0 on AIX
- IBM solidDB 7.0 on HP-UX
- IBM solidDB 7.0 on Linux
- IBM solidDB 7.0 on Solaris
- IBM solidDB 7.0 on Windows

Related concepts:

2.3.3, “solidDB installation requirements,” on page 38

2.3.4, “InfoSphere CDC for solidDB system requirements,” on page 40

2.3.2 Supported backend data servers for Universal Cache

The Universal Cache capability supports a number of IBM and other data servers as the backend data server.

IBM DB2 for Linux, UNIX, and Windows

- DB2 V9.8
- DB2 V9.7
- DB2 V9.5
- DB2 V9.1

IBM DB2 for iSeries

- DB2 for i/OS V6R1
- DB2 for i/OS V5R4

IBM DB2 for z/OS

- DB2 for z/OS V10
- DB2 for z/OS V9
- DB2 for z/OS V8

IBM Informix

- Informix V11.70
- Informix V11.50.3

The following Informix editions are supported:

- Informix Developer Edition
- Informix Ultimate Edition
- Informix Ultimate Warehouse Edition

For more information, see Informix product editions.

Oracle Database

- Oracle Database 11g
- Oracle Database 10g
- Oracle Database 9g

Microsoft SQL Server

- Microsoft SQL Server 2008
- Microsoft SQL Server 2005
- Microsoft SQL Server 2000

Sybase Adaptive Server Enterprise (ASE)

- Sybase ASE V15
- Sybase ASE V12.5.4

2.3.3 solidDB installation requirements

Before you install solidDB server, ensure that the system you choose meets the following software and disk and memory requirements.

- About 48 MB of disk space, including the space for separately installed documentation – the number varies considerably, depending on the platform
- At least 40 MB of RAM in the default configuration
- Adequate disk space for your database – an empty database typically requires about 16 MB of disk space
- If you use in-memory tables, additional memory to store those tables
- If you use InfoSphere CDC technology (or, the solidDB log reader is enabled), enough disk space to accommodate transaction log files preserved for replication recovery (catchup) – by default, the required log retention space is 10 GB
- Java Runtime Environment (JRE) or Java Development Kit (JDK), version 1.4.2 or newer, is required for
  - solidDB installation program

    Note: On Linux systems, the installation program does not support GNU Compiler for Java (GCJ).
  - Shared memory access (SMA) and linked library access (LLA) with Java

User process resource limits (`ulimits`) considerations in Linux and UNIX environments

In Linux and UNIX environments, you might need to modify the settings for the user process resource limits (**`ulimits`**) of your system. For details, see *OS user limit requirements (Linux and UNIX)*.

Security-enhanced Linux considerations

On Red Hat Enterprise Linux (RHEL) operating systems, if Security-enhanced Linux (SELinux) is enabled and in enforcing mode, the installer might fail because of SELinux restrictions.

To determine whether SELinux is installed and in enforcing mode, complete one of the following actions:

- Check the `/etc/sysconfig/selinux` file.
- Run the **`sestatus`** command.
- Check the `/var/log/messages` file for SELinux notices.

To disable SELinux, complete one of the following actions:

- Set SELinux in permissive mode and run the **`setenforce 0`** command as a superuser.
- Modify `/etc/sysconfig/selinux` and restart the computer.

If the solidDB server installs successfully on an RHEL system, all solidDB processes will run in the unconfined domain. To assign the processes to their own domains, so that also confined users can run them, you must modify the policy modules.

2.3.4 InfoSphere CDC for solidDB system requirements

Disk space requirements

Table 14. Disk space requirements

Disk space
InfoSphere CDC source system: • **100 GB**—Default value for the **Staging Store Disk Quota** for each instance of InfoSphere CDC. Use the InfoSphere CDC configuration tool to configure disk space for this quota. • **5 GB**—For installation files, data queues, and log files. • **Global disk quota**—Disk space is required on your source system for this quota which is used to store in-scope change data that has not been committed in your database. The amount of disk space required is determined by your replication environment and the workload of your source database. Use the **mirror_global_disk_quota_gb** system parameter to configure the amount of disk space used by this quota. InfoSphere CDC target system: • **1 GB**—The minimum amount of disk space allowed for the **Staging Store Disk Quota** for each instance of InfoSphere CDC. The minimum value for this quota is sufficient for all instances created on your target system. Use the InfoSphere CDC configuration tool to configure the disk space for this quota. • **5 GB**—For installation files, data queues, and log files. • **Global disk quota**—Disk space is required on your target system for this quota which is used to store LOB data received from your InfoSphere CDC source system. The amount of disk space required is determined by your replication environment and the amount of LOB data you are replicating. To improve performance, InfoSphere CDC will only persist LOB data to disk if RAM is not available on your target system. Use the **mirror_global_disk_quota_gb** system parameter to configure the amount of disk space used by this quota.

InfoSphere CDC may require additional disk space in the following situations:

- You are running large batch transactions in the database on your source system.
- You are configuring multiple subscriptions and one of your subscriptions is latent. In this type of scenario, InfoSphere CDC on your source system may persist transaction queues to disk if RAM is not available.
- You are replicating large LOB data types.
- You are replicating "wide" tables that have hundreds of columns.
- You are performing regular back ups of your metadata with the **dmbackupmd** command-line utility.

RAM requirements

Table 15. RAM requirements

RAM
Each instance of InfoSphere CDC requires memory for the Java Virtual Machine (JVM). The following default values for memory are assigned: • **1024 MB of RAM**—Default value for each 64-bit instance of InfoSphere CDC. • **512 MB of RAM**—Default value for each 32-bit instance of InfoSphere CDC. Use the InfoSphere CDC configuration tool to configure the memory for each instance of InfoSphere CDC. **Note:** InfoSphere CDC is predominantly a Java-based application. However, some portions of it are written in C. These portions of InfoSphere CDC are not subject to the memory limits specified for the JVM.

Although InfoSphere CDC memory requirements will fluctuate, you must work with your system administrator to ensure the allocated memory for each instance of the product is available at all times. This may involve deployment planning since other applications with memory requirements may be installed on the same server with InfoSphere CDC. Using values other than the defaults or allocating more RAM than is physically available on your server should only be undertaken after considering the impacts on product performance.

InfoSphere CDC source deployments may require additional RAM in the following scenarios:

- You are replicating large LOB data types with your InfoSphere CDC source deployment. These data types are sent to target while being retrieved from the source database. The target waits until all LOBs (for each record) are received before applying a row. LOBs are stored in memory as long as there is adequate RAM, otherwise they are written to disk on the target.
- You are replicating "wide" tables with hundreds of columns.
- You are performing large batch transactions in your source database rather than online transaction processing (OLTP).

Port requirements

InfoSphere CDC requires that you allocate a set of ports for communications with other components in the replication environment. The ports must be accessible through firewalls, although you do not require access to the internet.

Table 16. Port requirements

Protocol	Default port	Purpose
TCP	11101	Accepts connections from: • Management Console • Other installations of InfoSphere CDC as a source of replication • Command line utilities

3 Universal Cache evaluation setup overview

In the simplest Universal Cache evaluation setup, all components are installed on a single computer, except for the backend data server, which you should already have installed and running on a dedicated server machine.

Before you begin

1. Define the data to be cached.

   For evaluation purposes, it is assumed that you have an existing (backend) database that contains the data you want to cache. Ensure that the backend database is installed and running on a dedicated server machine. The Universal Cache components will be setup on a single evaluation computer that has a connection to the backend server machine.
2. Ensure that you have administrator access rights to your database (if one exists) and to all the computers or servers you will be installing components on.
3. Ensure that you have access to all the installation programs. See 2.2, "Component and installation package information," on page 24 for details.

About this task

Evaluation topology

A typical evaluation setup uses a two-node configuration:

- Evaluation node
  - Cache
    - solidDB server
    - InfoSphere CDC for solidDB replication engine
  - InfoSphere CDC Access Server
  - InfoSphere CDC for backend replication engine
  - solidDB ODBC Driver or solidDB JDBC Driver
  - InfoSphere CDC Management Console
- Database node
  - Backend data server (prerequisite)

Tip: Your evaluation configuration does not need to use the two-node configuration described above. Instead, you can install the components in several different configurations, as described in 2.1, "Installation topologies for Universal Cache," on page 20.

Setting up caching

Typically you would already have a working installation of the backend data server that contains the data you want to cache into the solidDB database. Setting up caching then includes defining the connection between the cache database and the backend database, defining which tables to cache, populating the cache database, and finally starting replication between the cache and the backend database.

Preparing applications for use with Universal Cache

You can setup the Universal Cache system for evaluation purposes without an existing application and use, for example, the backend and solidDB command line tools to issue simple SQL statements.

If you want to use an existing application that you have running against the backend database, the solidDB server provides various features that enable you to migrate your application to the Universal Cache environment with minimal changes. In the simplest scenario, you only need to modify the connection string to use the solidDB JDBC Driver or solidDB ODBC Driver.

Procedure

The installation and configuration of a Universal Cache system for evaluation purposes includes the following high-level steps:

1. Download and extract the installation files to the evaluation node.
2. Install and configure the Universal Cache components using the installation wizards provided with each component.
3. Set up caching of data between your backend data server and solidDB server using Management Console. During the setup, you can create and populate tables in the solidDB database with data from the backend tables.
4. Activate Universal Cache by starting replication between the cache and the backend database.

4 Installing and configuring Universal Cache for evaluation setup

This section provides a high-level overview of the installation and configuration steps for Universal Cache, when setting up the environment for evaluation purposes.

Prerequisites for evaluation setup

The installation and configuration instructions for evaluation purposes assume the following:

- You have a working installation of the backend server with a database that contains data that you want to replicate to and from solidDB server.
- Your configuration includes only one solidDB server.

Install and configure procedure overview

Important: Install the components in the order described below; this is to ensure that you meet installation and configuration requirements for each component.

1. **Locate the installation images for the Universal Cache components**.

   For a list of the installation images for different platforms, see 2.2, "Component and installation package information," on page 24.
2. **Ensure that you have access to all the following documentation packages that are needed when installing Universal Cache.**
   - IBM solidDB 7.0 Information Center or *IBM solidDB 7.0 Documentation* package in PDF format
   - IBM InfoSphere Change Data Capture version 6.5 Information Center or *InfoSphere Change Data Capture Documentation* package in PDF format
3. **Ensure that you have system administrator (or equivalent) access rights to all the nodes where you will install the Universal Cache components.**

   Tip: While setting up Universal Cache, you need to create (or use existing) user accounts, database, and network connection identification data to enable the different components to communicate with each other.

   A summary of the key identification data is available in User accounts and database connection data for Universal Cache.
4. **Install and configure IBM solidDB server.**

   For details, see Installing and configuring solidDB server for Universal Cache.

   Result: You have a working solidDB server installation with an empty database.
5. **Install and configure InfoSphere CDC for IBM solidDB**.

   For details, see Installing and configuring InfoSphere CDC for solidDB.

   Result: You have a working installation of the replication engine and you have created at least one InfoSphere CDC instance that is connected to your solidDB database.
6. **Install and configure InfoSphere CDC for the backend data server.**

   For details, see Installing and configuring InfoSphere CDC for your backend data server.

Result: You have a working installation of the replication engine and you have created at least one InfoSphere CDC instance that is connected to your backend database.

7. **Install InfoSphere CDC Access Server.**

   For details, see Installing and configuring Access Server.

   Result: You have a working Access Server installation and you have created a system administrator account for logging into the Management Console.

8. **Install InfoSphere CDC Management Console.**

   For details, see Installing and configuring Management Console.

   Result: You have a working Management Console installation and you can login into the InfoSphere CDC Management Console using the system administrator account.

9. **Set up the replication subscriptions.**

   For details, see 5, "Setting up caching with Management Console," on page 61.

   Result: You have created replication subscriptions between your solidDB and backend data servers.

4.1 User accounts and database connection data for Universal Cache

When installing and configuring Universal Cache, you need to create or use existing user accounts and database and connection information to enable the different components to communicate with each other. The table in this section summarizes the user accounts and database connection data that are created when setting up Universal Cache.

The default values are given if available.

solidDB

Table 17. User account and network connection data for solidDB

solidDB	Example value (default if available)	Usage
Server connection data (server name and port number)	tcp 1964	• Defined in `solid.ini` configuration file • Needed when creating InfoSphere CDC for solidDB instances
Database login data	Username: soliduser Password: admsolid	• Defined when creating the solidDB database • Needed when creating InfoSphere CDC for solidDB instances
System catalog name	DBA	• Defined when creating the solidDB database The solidDB syntax for database object hierarchy is the following: `catalog_name.schema_name.database_object` For more details, see section *Managing database objects* in *IBM solidDB SQL Guide*. **Important:** Subscriptions can only include tables that are included in the System catalog.
Schema name	SOLIDUSER	• The default schema name is the username. You can create new schemas using the `CREATE SCHEMA` statement. • Needed when creating InfoSphere CDC for solidDB instances

InfoSphere CDC for solidDB

Table 18. User account and network connection data for InfoSphere CDC for solidDB

InfoSphere CDC for solidDB	Example value (default if available)	Usage
Instance Name	solid-inst	• Defined when creating the InfoSphere CDC instance • Used when administering the instance with `dm` commands.
Server Port	11101 (default)	• Defined when creating the InfoSphere CDC instance • Needed when connecting to the instance from the Management Console / Access Manager
Windows Service user account		• Defined when creating the InfoSphere CDC instance • Needed when administering InfoSphere CDC services (for example, starting the instance)
Database login data	Username: soliduser Password: admsolid Metadata schema: SOLIDUSER	• Specifies the login data to the solidDB database and the schema name that is used for InfoSphere CDC metadata tables
Server connection data	cache-node 1964	• Specifies the connection data to the solidDB server • The host name can be given as the network name or IP address. If the InfoSphere CDC replication engine is located on the same node as the solidDB server, the host name can also be `localhost`. • The port number must be a port that the solidDB server is listening to (defined in the `solid.ini` configuration file)

InfoSphere CDC for backend data server

Table 19. User account and network connection data for InfoSphere CDC for backend data server

InfoSphere CDC for backend data server	Example value (default if available)	Usage
Instance Name	BE-inst	• Defined when creating the InfoSphere CDC instance • Used when administering the instance with `dm` commands.
Server Port	10901 (default depends on the backend data server)	• Defined when creating the InfoSphere CDC instance • Needed when connecting to the instance from the Management Console / Access Manager
Windows Service user account		• Defined when creating the InfoSphere CDC instance • Needed when administering InfoSphere CDC services (for example, starting the instance)
Database login data	Depends on the backend data server	• Specifies the login data and connection settings for your backend database For details, check the section *Before your install*: *Required database, user accounts, and schemas* in the *InfoSphere Change Data Capture, End-User Documentation* for your backend data server.

Access Server

Table 20. User account and network connection data for Access Server

Access Server	Example value (default if available)	Usage
Port Number	10101 (default)	• Defined when installing (Windows) or configuring (Linux and UNIX) the Access Server • Needed when login in to the Management Console
Login data (System Administrator)	Username: admin (default) Password: uc123	• Defined when installing (Windows) or configuring (Linux and UNIX) the Access Server • Specifies the username for the Access Server System Administrator • Needed when login in to the Management Console

Management Console

Table 21. User account and network connection data for Management Console

Management Console	Example value (default if available)	Usage
Login data (System Administrator)	Username: admin (default) Password: uc123	• Defined when installing (Windows) or configuring (Linux and UNIX) the Access Server • Specifies the username for the Access Server System Administrator
Server Name		• Specifies the host name (system name) or full IP address of the workstation running Access Server. • Used my Management Console to connect to the Access Server
Port Number	10101 (default)	• Defined when installing (Windows) or configuring (Linux and UNIX) the Access Server

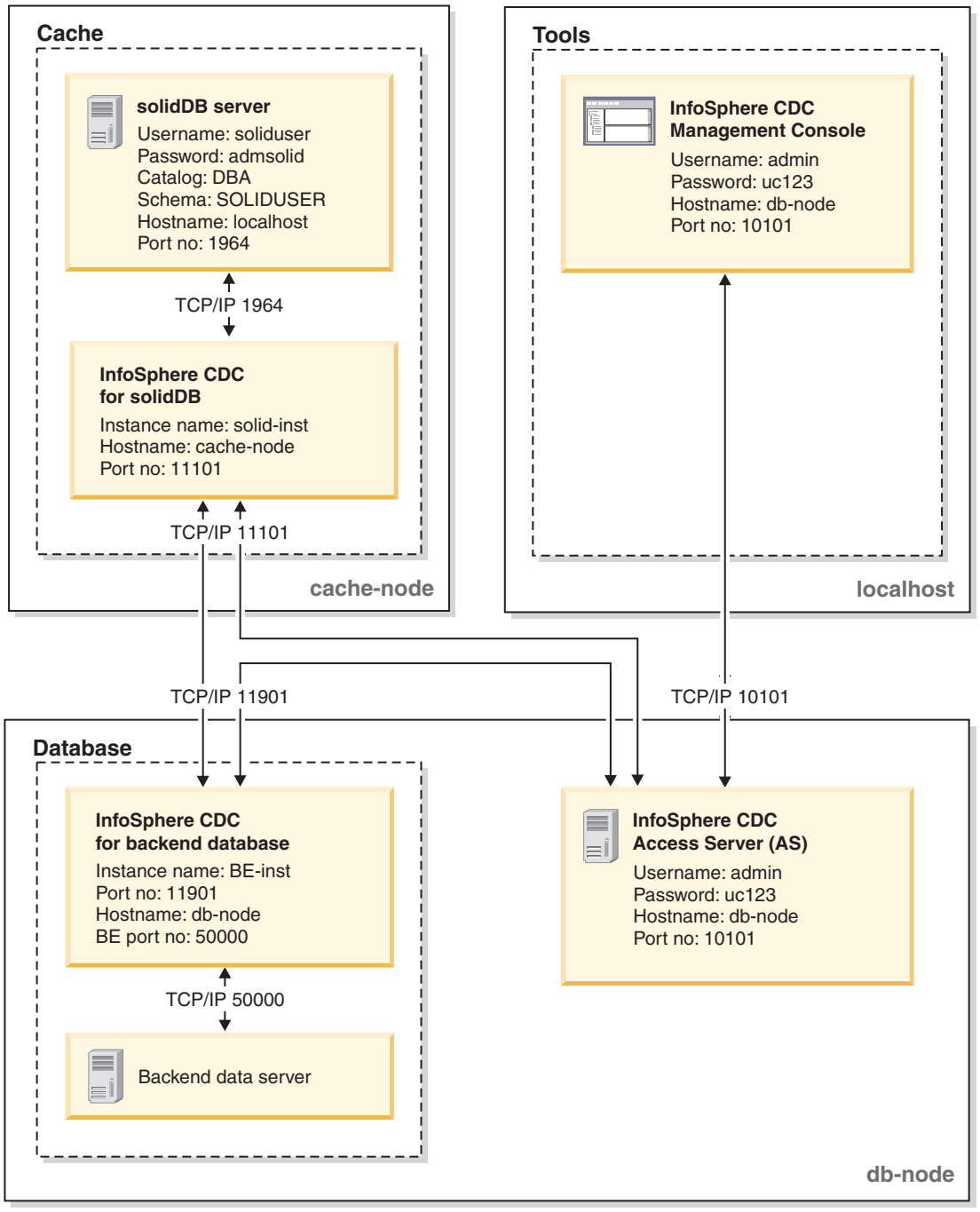

Figure 12. Example: User accounts and database connection data for Universal Cache

4.2 Installing and configuring solidDB server for Universal Cache

4.2.1 Installing solidDB server for Universal Cache

Procedure

1. Install Java Runtime Environment (JRE) or Java Development Kit (JDK), version 1.4.2 or newer, if not already installed.

   JRE or JDK 1.4.2 or newer is needed to run the solidDB installer.

Note: On Linux systems, GNU Compiler for Java (GCJ) is not supported.

2. On the downloaded installation image or the installation DVD, locate the installation program file for your operating system:
   - `solidDB-7.0-<platform>.exe` (Windows)
   - `solidDB-7.0-<platform>.bin` (Linux and UNIX)
3. Double-click the installation program file. The solidDB installation wizard starts.
4. Follow the instructions on the wizard to complete the installation.

   Note: In Linux and UNIX operating systems, you must be able write to the directory that you are using for the installation. If the installation program cannot create the directory, you are prompted to specify a different directory.
5. Verify your solidDB server installation and familiarize yourself with the basic administration tasks.

   For more information, see the section *Verifying solidDB installation* in the *IBM solidDB Getting Started Guide* , as well as the *IBM solidDB Administrator Guide*.

What to do next

4.2.2, “Configuring solidDB server for Universal Cache capability”

4.2.2 Configuring solidDB server for Universal Cache capability

To be able to use the solidDB server with the InfoSphere CDC technology, you need modify configuration settings so that the InfoSphere CDC for solidDB can connect to and replicate data from your solidDB database.

Before you begin

This section assumes that you are familiar with solidDB administration and have read, for example, the sections *Administering solidDB* and *Configuring solidDB* in the *IBM solidDB Administrator Guide*.

Procedure

1. **Set up your database environment by creating a working directory, your solidDB database, and user accounts.**

   For instructions, see *Creating a new database* in the *IBM solidDB Administrator Guide*.

   Tip:

   After you have installed the solidDB server, you can find the following directories in the installation directory:

```
<installation directory>
    bin\
    ..
    eval_kit\
        standalone\
        cdc\
    ..
    samples
    ..
```

You can use the `eval_kit/cdc` directory in the solidDB server installation directory as your working directory; it contains a sample `solid.ini` file for using solidDB with the Universal Cache capability or InfoSphere CDC replication.

2. **Configure the Log Reader by modifying the configuration parameters in the `LogReader` section of the `solid.ini` configuration file.**
   a. Set the **`LogReader.LogReaderEnabled`** parameter to `yes`.

      ```
      [LogReader]
      LogReaderEnabled=yes
      ```

      You must enable the Log Reader to be able to use the solidDB database as a source database in InfoSphere CDC replication. The factory value of the **`LogReader.LogReaderEnabled`** parameter is `no`.

   b. Set the transaction log retention space size with the **`LogReader.MaxLogSize`** parameter.

      ```
      [LogReader]
      MaxLogSize=<MB>
      ```

      The **`LogReader.MaxLogSize`** parameter sets the amount (size) of log files that are available for performing a catchup. The maximum size of log file depends on the available disk space and the expected downtime after which a catchup is needed. The factory value is 10 240 (10 GB).

      If the log reader is enabled, the specified log file retention space is always used fully. Also, the log files can use even more space, if backups are not performed or the parameter **`General.CheckpointDeleteLog`** is set to `no`.

   c. Set the in-memory buffer size for log records with the **`LogReader.MaxSpace`** parameter.

      ```
      [LogReader]
      MaxSpace=<number of log records>
      ```

      The **`MaxSpace`** parameter sets the size (in number of log records) of the in-memory log reader buffer used in throttling. The maximum number of log records depends on the expected size of load bursts. The factory value is 100000 log records.

      The size of a log record is that of the (binary) row size, plus a few bytes of additional metadata overhead. When the buffer fills up, throughput throttling is applied; the operations are blocked until there is room in the log reader buffer.

3. **Modify other performance and database-setup-related configuration parameters as necessary.**
   - **`Logging.DurabilityLevel`**

     By default, the solidDB server durability level is set to relaxed (**`Logging.DurabilityLevel=1`**). Relaxed durability means that most recent transactions can be lost if the server fails unexpectedly.

     To prevent loss of data, set the durability level to strict with the following setting in the `solid.ini` file:

     ```
     [Logging]
     DurabilityLevel=3
     ```

     Note: Strict durability setting induces a performance penalty when compared to relaxed durability. Relaxed durability can be used without the risk of data loss if solidDB HA (HotStandby) configuration is applied with the 2-Safe replication protocol (default).

   - **`General.DefaultStoreIsMemory`**

By default, the solidDB table storage type is set to M-table (**General.DefaultStoreIsMemory=yes)**.

- **Sql.IsolationLevel**

  By default, the solidDB isolation level is set to Read Committed (**Sql.IsolationLevel=1**).

4.3 Installing and configuring InfoSphere CDC for solidDB

To install InfoSphere CDC for solidDB, follow the instructions in the installation wizard. After installation, use the InfoSphere CDC configuration tool to configure your InfoSphere CDC for solidDB instances.

Before you begin

Ensure that:

- Your solidDB server is running.
- You have created your solidDB database.
- You know the username and password for your solidDB database.
- You know the network address and port number the solidDB server is listening to.
- You have created a new schema or decided which existing schema you want InfoSphere CDC for solidDB to create the metadata tables in.

Procedure

1. **Install InfoSphere CDC for solidDB.**
   a. On the downloaded installation image or the installation DVD, locate the installation program file for your operating system:
      - `setup-x86-solid.exe` (Windows)
      - `setup-<platform>-solid.bin` (Linux and UNIX)
   b. Double-click the installation program file. The installation wizard starts.
   c. Follow the wizard's instructions to complete the installation.

      Note: In Linux and UNIX operating systems, you must be able write to the directory that you are using for the installation. If the installation program cannot create the directory, you are prompted to specify a different directory.

   At the end of installation, select to launch the InfoSphere CDC configuration tool to configure your InfoSphere CDC for solidDB instances.
2. **Using the configuration tool, create a new instance of InfoSphere CDC for solidDB.**
   - 4.3.1, “To add a new instance of InfoSphere CDC (Windows),” on page 53
   - 4.3.2, “To add a new instance of InfoSphere CDC (UNIX and Linux),” on page 56

What to do next

Continue with 4.4, “Installing and configuring InfoSphere CDC for your backend data server,” on page 58.

4.3.1 To add a new instance of InfoSphere CDC (Windows)

Procedure

1. If you are configuring the first instance of InfoSphere CDC after installation, you can proceed to Step 3 of this procedure.
2. At the command prompt, launch the configuration tool by issuing the following command in the specified directory:

   ```
   \<InfoSphere CDC Installation Directory>\bin\dmconfigurets
   ```

3. At the welcome message, click **OK** to continue.
4. On the **IBM InfoSphere CDC New Instance** dialog box, you can configure the following options in the **Instance** area:

Option	Description
Name	Type a name for your InfoSphere CDC instance. This name must be unique.
Server Port	Type the port number which InfoSphere CDC uses for communication with client workstations running Management Console and other servers. **Note:** This port number cannot be used by other applications installed on the same server. You will use this port number when specifying access parameters for your datastore in the Access Manager perspective in Management Console. InfoSphere CDC displays a default TCP/IP port of 11101. For more information, see your Management Console documentation. **Note:** If you install several instances on the same node, the port number for each instance must be unique.
Auto-Discovery Port	Select the box and type the UDP port number that is used for auto-discovery broadcasts sent from Access Server. For more information about auto-discovery, see your Management Console documentation.
Maximum Memory Allowed	Type the maximum amount of RAM you want to allocate for InfoSphere CDC. You must allocate a minimum of 64 MB for each instance you configure. By default, there is 512 MB of RAM allocated for a 32 bit instance and 1024 MB of RAM allocated for a 64 bit instance.
Staging Store Disk Quota (GB)	Enter the maximum amount of disk space that will be utilized by the InfoSphere CDC staging store on your source system. The default value is 100 GB and minimum value is 1 GB. Specify 1 GB if you are creating an instance that will be used as a target of replication. This reduces the disk resources that InfoSphere CDC requires on your target system.

Option	Description
Bit-Version	Select the bit-version of your database by selecting one of the following options: • 32 bit • 64 bit These options are not enabled if you are installing InfoSphere CDC on a 32-bit server.

5. In the **Windows Service** area, you can specify the account that will be used to start InfoSphere CDC services. Select one of the following options:

Option	Description
Local System account	Start InfoSphere CDC services through the local system administrator account.
This account	Start InfoSphere CDC services through the specified user account. The account must be specified in the format *<domain>\<user name>*, where *<domain>* is the name of a domain in your environment, and *<user name>* is a valid login user name in the specified domain. If your computer is not part of a domain, you can specify *<computer name>\<user name>*. In the **Password** and **Confirm Password** boxes, type the password currently associated with the selected Windows user account. If you change the password for the Windows user account after installing InfoSphere CDC, you will have to use the Windows Services dialog to change the password currently set for each InfoSphere CDC service.

6. In the **Database** area, you can configure access to the database that contains the tables for replication. To complete this step, you will require system administrator privileges. You can then add a datastore in the Access Manager perspective in Management Console and provide users access to this database. For more information, see your Management Console documentation.

Option	Description
User name	Type the user name for the specified database.
Password	Type the password for the specified database.

Option	Description
Metadata Schema	Select the schema for the database that will be used for the InfoSphere CDC metadata tables. As a default, the user name entered above is used. You can specify any schema except those in use by other installed instances of InfoSphere CDC for the given database. You must set up or decide on this schema as part of the installation prerequisites. **Note:** Make sure to use UPPERCASE letters for the metadata schema. By default, all the schema names (catalog names) in solidDB are in uppercase.
Advanced	The **Advanced** button enables you to modify configuration parameters for the solidDB JDBC driver. For more information about the JDBC driver parameters, see the *IBM solidDB Programmer Guide*. **Tip:** • In HA setup, the parameter **`solid_tf_level`** is by default set to `CONNECTION`. • In SMA setup, the parameter **`solid_shared_memory`** is by default set to `yes`. • To enable use of operating-system-based external authentication, set the following properties: – solid_use_strong_encryption=yes – solid_gskit_path=*location_of_GSKit_library* **Important:** To authenticate users using the operating-system-based authentication mechanisms, the IBM Global Security Kit (GSKit) must be enable on both the server and client computers.

7. In the **Server** area, you can configure the solidDB server that you want to replicate data to or from and which contains all of the tables for replication. You can configure either single server or HA configuration (HotStandby).

Option	Description
Single server	Type the host name and port number for the specified solidDB server.
Enable SMA	Select the check box if you are using solidDB with shared memory access (SMA).
HA Configuration (HotStandby)	Type the host names and port numbers for the specified Primary and Secondary solidDB servers.

8. Click **OK** to save your configuration settings for the InfoSphere CDC instance.
9. If InfoSphere CDC has detected an unsupported encoding, a dialog will open asking you to select an alternate encoding from a list.

You can filter the list of alternate encodings by clicking one of the following buttons:

- **Closest match**—Displays the alternated encodings that are the closest match to the data.
- **Comparable encodings byte length**—Displays the alternate encodings in order of byte length.
- **All**–Displays all alternate encodings.

Select an encoding from the list and click **OK**.

If you click **Cancel**, an error message will be displayed and the instance will not be created.

What to do next

After you complete the configuration, you can start InfoSphere CDC.

4.3.2 To add a new instance of InfoSphere CDC (UNIX and Linux)

Procedure

1. If you are configuring the first instance of InfoSphere CDC after installation, you can proceed to Step 3 of this procedure.
2. At the command prompt, launch the configuration tool by issuing the following command in the specified directory:
   `/<InfoSphere CDC Installation Directory>/bin/dmconfigurets`
3. At the welcome message, press **Enter** to continue.
4. Type 2 and press **Enter** to add a new instance of InfoSphere CDC.
5. Type a name for your InfoSphere CDC instance and press **Enter**. The instance name must be unique.
6. Type the port number which InfoSphere CDC uses for communication with client workstations running Management Console and other servers. InfoSphere CDC displays a default port of `11101`. Press **Enter**.

   Note: This port number cannot be used by other applications installed on the same server. You will use this port number when specifying access parameters for your datastore in the Access Manager perspective in Management Console. For more information, see your Management Console documentation.

   Note: If you install several instances on the same node, the port number for each instance must be unique.
7. If you are using the auto-discovery feature in Access Manager, then enable the this feature by typing the UDP port number that you set in Access Server. InfoSphere CDC uses this UDP port number for auto-discovery broadcasts sent from Access Server. Otherwise, press **Enter** to disable this feature.
8. Type the amount of physically available RAM you want to allocate for InfoSphere CDC. You must allocate a minimum of 64 MB for each instance you configure. By default, there is 512 MB of RAM allocated for a 32 bit instance and 1024 MB of RAM allocated for a 64 bit instance.
9. Select the solidDB server configuration type you want to configure.

Option	Description
Single server	Type 1 and press **Enter**.
HA Configuration (HotStandby)	Type 2 and press **Enter**.

10. Type the host name and port number according to your configuration type.

Option	Description
Single server	1. Type the host name for the specified server and press **Enter**. 2. Type the port number for the specified server and press **Enter**. Default is 1964.
HA Configuration (HotStandby)	1. Type the host name for the specified Primary server and press **Enter**. 2. Type the port number for the specified Primary server and press **Enter**. Default is 1964. 3. Type the host name for the specified Secondary server and press **Enter**. 4. Type the port number for the specified Secondary server and press **Enter**. Default is 1964. **Note:** The default port number for Primary and Secondary is the same as it is assumed that the Primary and Secondary are located on different nodes. If, for example, for evaluation purposes, your Primary and Secondary servers are located on the same node, the default port number for both cannot be the same.

11. Select to enable the use of solidDB with shared memory access (SMA) as necessary.

Option	Description
Use default settings	Type `n` and press **Enter**.
Enable SMA	Type `y` and press **Enter**.

12. Configure advanced parameters (JDBC parameters) as necessary.

Option	Description
Use default settings	Type `n` and press **Enter**.
Modify settings	1. Type `y` and press **Enter** 2. Enter the parameter settings using the syntax `<parameter>=<value>;<parameter>=<value>;...` **Tip:** • In HA setup, the parameter **`solid_tf_level`** is by default set to `CONNECTION`. • In SMA setup, the parameter **`solid_shared_memory`** is by default set to `yes`. • To enable use of operating-system-based external authentication, set the following properties: – solid_use_strong_encryption=yes – solid_gskit_path=*location_of_GSKit_library* **Important:** To authenticate users using the operating-system-based authentication mechanisms, the IBM Global Security Kit (GSKit) must be enable on both the server and client computers.

13. Type the user name for the specified database and press **Enter**.
14. Type the password for the specified database and press **Enter**. The configuration tool will now search the database for schemas.
15. Type the number that corresponds to the metadata schema that you would like to use and press **Enter**.
16. Type the path to the directory that will be used for bulk inserts into the database. Press **Enter**. Both your solidDB database and InfoSphere CDC must have read and write permissions for this directory.

    Notes:
    - You should use a different directory for each instance of InfoSphere CDC.
    - This directory may contain database tables for replication. You should take this into consideration when determining user access to this directory.
17. If InfoSphere CDC detects an unsupported encoding, an error message will be displayed and you will be asked to choose an alternate encoding.
    a. Enter `y` to proceed.

       Note: If you enter `n` and press **Enter** to cancel, the instance will not be created.
    b. Enter a value to choose how the alternate encodings will be displayed:
       - **1**—Displays the available alternate encodings that are the closest match to the database.
       - **2**—Displays the available alternate encodings in order of byte length.
       - **3**—Displays all available alternate encodings.
    c. Enter the number for the encoding to be used and press **Enter**.
18. The configuration tool creates the InfoSphere CDC instance and prompts you to start the instance. Type `y` to start the instance.

    Note: The configuration tool will prompt you if your configuration is about to overwrite the metadata for an existing instance.

4.4 Installing and configuring InfoSphere CDC for your backend data server

To install InfoSphere CDC for your backend data server, follow the instructions in the installation wizard. After installation, use the InfoSphere CDC configuration tool to configure your InfoSphere CDC instances.

Before you begin

- Check that your backend data server is running.
- You have created your backend database.
- You know the username and password for your backend database.
- You know the network address and port number the backend data server is listening to.
- You have created a new schema or decided which existing schema you want InfoSphere CDC to create the metadata tables in.

Procedure

1. **Check the installation prerequisites.**

The installation requirements are described in section *Before you install* in the *InfoSphere Change Data Capture, End-User Documentation* for your backend data server.

2. **Install InfoSphere CDC for the backend data server.**

   For detailed instructions, see section *Installing InfoSphere CDC* in the *InfoSphere Change Data Capture, End-User Documentation* for the backend data server. At the end of installation, select to launch the InfoSphere CDC configuration tool to configure your InfoSphere CDC instances.

3. **Using the configuration tool, create a new instance of InfoSphere CDC for the backend data server.**

   For detailed instructions, see section *Configuring InfoSphere CDC* in the *InfoSphere Change Data Capture, End-User Documentation* for the backend data server.

   Tip: If you intend to use bidirectional replication and your backend data server is DB2 for Linux, UNIX, and Windows, set the InfoSphere CDC for DB2 system parameter **ddl_awareness** to `false`.

What to do next

4.5, “Installing and configuring InfoSphere CDC Access Server”

4.5 Installing and configuring InfoSphere CDC Access Server

To install the Access Server, follow the instructions in the installation wizard. After installation, if your network uses a firewall or other security mechanism that requires static ports for communication, you must specify the ports that other computers can use to communicate with Access Server services.

Procedure

1. Install Access Server according to the instructions in the InfoSphere Change Data Capture Access Server and Management Console, Installation Guide.

   Important: The Access Server account is created during the installation. The Access Server account is used for the following operations:

   - Logging in to Management Console
   - Managing users and datastores in Management Console

2. If necessary for your environment, specify the ports that other computers can use to communicate with Access Server services.

   For instructions, see section *After you install Access Server* in the InfoSphere Change Data Capture Access Server and Management Console, Installation Guide.

What to do next

4.6, “Installing and configuring InfoSphere CDC Management Console”

4.6 Installing and configuring InfoSphere CDC Management Console

To install the Management Console, follow the instructions in the installation wizard. After installation, log in to the Management Console using the system administrator account that you created when installing the Access Server.

Procedure

1. Install Management Console according to the instructions in InfoSphere Change Data Capture Access Server and Management Console, Installation Guide.
2. Log in to the Management Console with the system administrator account that you created when installing the Access Server.

What to do next

- View the Management Console help documentation through the **Help** > **Help Contents** menu path.
- Continue setting up Universal Cache.

5 Setting up caching with Management Console

The InfoSphere CDC Management Console is an interactive GUI tool that you can use to configure and monitor replication (caching) *subscriptions* between the cache and backend databases. This section provides a high-level overview of how you can create replication subscriptions for evaluation purposes.

Before you begin

- Check that the tables you intend to replicate exist in the backend database.
- Check that your solidDB and backend databases are running.
- Check that your InfoSphere CDC instances for solidDB and backend data servers are running.
- Check that you have sufficient access privileges to your databases.
- Check that you have defined your desired replication principles in accordance with your business rules.
- If you intend to use bidirectional replication and your backend data server is DB2 for Linux, UNIX, and Windows, set the InfoSphere CDC for DB2 system parameter **ddl_awareness** to `false`.

Procedure

1. **Log in to Management Console by connecting to Access Server.**

   For more details, see section *Logging into Management Console (Connecting to Access Server)* in the *InfoSphere Change Data Capture Management Console, Administration Guide*.

   Tip: To be able to work in the Access Manager perspective of the Management Console, you must be a System Administrator that has the privilege to manage datastores and user accounts. The System Administrator account was created during the installation of the Management Console.
2. **Set up datastores for the solidDB and backend databases.**
   a. Add new datastore for the solidDB database.
      1) Click **Access Manager** > **Datastore Management**.
      2) Click **File** > **Access Server** > **New Datastore**.
      3) Type the name of the datastore in the **Name** box.
      4) Type a description in the **Description** box.
      5) In the **Host Name** box, type the host name or the full IP address of the server where you have installed InfoSphere CDC for solidDB.
      6) In the **Port** box, type the port number which InfoSphere CDC uses for communication with the other components. For example, InfoSphere CDC for solidDB uses by default port number 11101.
      7) Ping the server. If successful, this returns the datastore properties including the type of server where you have installed InfoSphere CDC and the version number of the product.
   b. Add new datastore for the backend database.
      1) Click **Access Manager** > **Datastore Management**.
      2) Click **File** > **Access Server** > **New Datastore**.
      3) Type the name of the datastore in the **Name** box.

4) Type a description in the **Description** box.
5) In the **Host Name** box, type the host name or the full IP address of the server where you have installed InfoSphere CDC.
6) In the **Port** box, type the port number which InfoSphere CDC uses for communication with the other components. For example, InfoSphere CDC for Informix uses by default port number 10901.
7) Ping the server. If successful, this returns the datastore properties including the type of server where you have installed InfoSphere CDC and the version number of the product.

c. Assign users to the datastores.

You need to assign the same users to both the solidDB datastore and the backend datastore.

1) Click **Access Manager** > **Datastore Management**.
2) Select a datastore.
3) Right-click and select **Assign User**.
4) Select a user or hold **Ctrl** to select multiple users.
5) Review the connection parameters. Click **OK** to accept the default connection parameters on the datastore or modify the parameters for the selected users.

For detailed instructions, see section *Setting up datastores* in the *InfoSphere Change Data Capture Management Console, Administration Guide*.

3. **Set up subscriptions.**

For detailed instructions on how to set up subscriptions using Management Console, see section *Setting up subscriptions* in the *InfoSphere Change Data Capture Management Console, Administration Guide*.

Tip: As an example, the following steps describe how to create subscriptions for bidirectional replication environment.

a. Create a new backend-to-solidDB subscription.
1) Click **Configuration** > **Subscriptions**.
2) Right-click anywhere in the **Subscriptions** field and select **New Subscription**.
3) Type the name of the new backend-to-solidDB subscription in the **Name** box.
4) Type the description of the new subscription in the **Description** box.
5) Select the backend datastore from the **Source** list.
6) Select the solidDB datastore from the **Target** list.
7) Click **OK**.

b. Create a new solidDB-to-backend subscription.
1) Click **Configuration** > **Subscriptions**.
2) Right-click anywhere in the **Subscriptions** field and select **New Subscription**.
3) Type the name of the new solidDB-to-backend subscription in the **Name** box.
4) Type the description of the new subscription in the **Description** box.
5) Select the solidDB datastore from the **Source** list.
6) Select the backend datastore from the **Target** list.
7) Click **OK**.

4. **Map tables for replication in both subscriptions.** This procedure assumes the backend data server contains the tables you want to cache into the solidDB database.
   a. Click **Configuration** > **Subscriptions**.
   b. Select the backend-to-solidDB subscription, right-click and select **Map Tables**.
   c. Select **Multiple One-to-One Mappings** and click **Next**.
   d. Expand the database, schema, or table from the **Source Tables** list to view tables from your database that are available for mapping. Right-click the database user or schema and click **Refresh** if you do not see your table listed.
   e. Enable one or more tables to map from the **Source Tables** list.
   f. Click **Next**.
   g. Click **Create new target tables**.
   h. Click **Next**.
   i. Specify a target owner for each source owner.
   j. Specify how the new target table names relate to their corresponding source table names.
   k. Click **Next**.
   l. Set the replication method to **Mirror (Change Data Capture)**.
   m. Verify the mappings in the **Complete Mappings** dialog, and click **Next**.
   n. Review the mapping summary and click **Finish**.

   For detailed instructions, see section *Mapping tables* in the *InfoSphere Change Data Capture Management Console, Administration Guide*.
5. Optional: **For each table mapping, set conflict detection and resolution in accordance with your business rules.**
   a. Click **Configuration** > **Subscriptions**.
   b. Select the subscription.
   c. Click the **Table Mappings** view and select the table mapping from the **Source Table** column.
   d. Right-click and select **Open Details....**
   e. Click the **Conflicts** tab.
   f. Select the columns on which you want to detect conflicts.
   g. Select the conflict resolution from the **Conflict Resolution Method** list.
   h. Click **Save**.

   For detailed instructions, see section *Setting conflict detection and resolution* in the *InfoSphere Change Data Capture Management Console, Administration Guide*.
6. Optional: **Set character set conversions for source columns.**

   If your solidDB database mode is Unicode (**`General.InternalCharEncoding=UTF8`**), set the encoding of character data type columns (CHAR, VARCHAR, and so on) to UTF-8.
7. **Start replication on subscriptions.** To start caching, start continuous mirroring on the subscriptions you have created.
   a. Click **Monitoring** > **Subscriptions**.
   b. Right-click on the two subscriptions and select **Start Mirroring**.
   c. Select **Continuous** and click **OK** to start mirroring.

For detailed instructions, see section *Starting and ending replication on subscriptions* in the *InfoSphere Change Data Capture Management Console, Administration Guide*.

Results

You have set up, for example, bidirectional replication subscriptions between the backend and solidDB databases. As you make changes in either database, InfoSphere CDC replication mechanism takes care of replicating the changes to the other database.

For example, you can use solidDB SQL Editor (**solsql**) to issue SQL statements in the solidDB server. The InfoSphere CDC components will then take care of replicating the changes to the backend database.

Part 3. Appendixes

Notices

© Copyright Oy IBM Finland Ab 1993, 2013.

All rights reserved.

No portion of this product may be used in any way except as expressly authorized in writing by IBM.

This product is protected by U.S. patents 6144941, 7136912, 6970876, 7139775, 6978396, 7266702, 7406489, 7502796, and 7587429.

This product is assigned the U.S. Export Control Classification Number ECCN=5D992b.

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, ibm.com, Solid, solidDB, InfoSphere, DB2, Informix, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other product and service names might be trademarks of IBM or other companies.

SC27-3848-05