Study Unit 5

Security and Recovery

Learning Outcomes

By the end of this unit, you should be able to:

1. Explain processing rights and privileges, and describe the DBMS security model.
2. Construct SQL/DCL statements to grant, revoke and deny permissions to user and roles.
3. Explain and apply security and recovery measures
4. Differentiate between reprocessing and recovery using rollback/rollforward
5. Propose recovery procedure to recover from a system failure
6. Revise recovery procedure with the introduction of checkpoints

Overview

A DBMS can determine whether users are performing legitimate operations on a database based on whether they have been given only those processing rights to perform the operations. Specific database operations include executing a stored procedure or creating a table. When a user is not given a processing right for specific database operation, the DBMS will not allow the operation.

In this study unit, we use SQL/DCL constructs: grant, revoke and deny permissions. The processing rights or permissions given to users are determined by the role they play. Permissions can also be withdrawn when a job scope is changed.

As data is an important resource in any organisation, it is of utmost importance to have a security plan in place. The study unit also discusses guidelines to secure the database server.

We next look at database recovery, another facility provided by the DBMS. Database operations requested by a user are first carried out on memory buffer. Between the time that a buffer is dirtied to the time that the buffer is written back to secondary storage, a failure may occur. Database recovery aims to restore the database to a consistent state is necessary.

The study unit covers two main recovery methods. Recovery via reprocessing is the simplest form of recovery method. The transactions are re-executed. Recovery via rollback/rollforward requires the transaction log file. Instead of re-executing the database operations, this recovery method uses the before and after images and simply applies them onto the database. We will also discuss checkpoints and how they reduce recovery time.

This study unit covers chapter 9 of the course text. It is estimated that the student will spend about 6 hours to read the course text chapter 9 in conjunction with the study notes, to work out the activities, and self-assessment questions in the study notes included at suitable junctures to test understanding of the contents covered. It is advisable to use the study notes to guide the reading of the chapter in the textbook, attempt the questions, and then check the text and/or other sources for the accuracy and completeness of your answers.

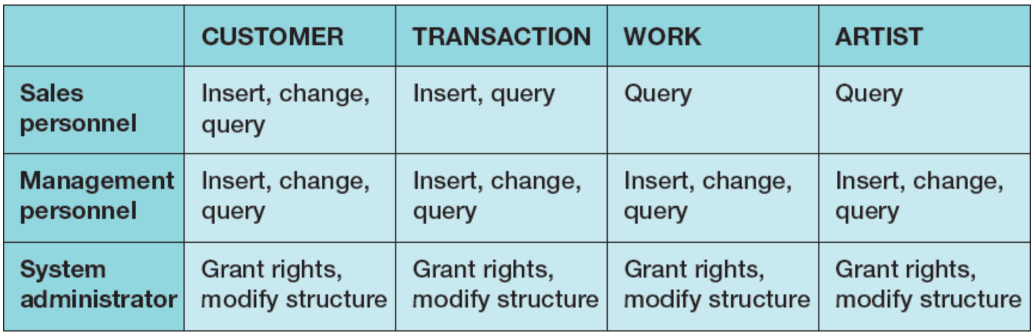
Chapter 1 Security

While concurrency control is about ensuring that that concurrent user work does not unintentionally interfere with one another's work, security is about ensuring that a user has been given permission to perform the requested database operation.

1.1 Processing Rights and Responsibilities

A DBMS allows us to write SQL Data Control Language (DCL) statements to give or remove processing rights. A processing right is a permission to allow a user to perform a specific database operation, e.g., to allow a user to execute a stored procedure or to create a table.

The processing rights given to users are determined by the role they play. For example, as shown in Figure 5.1, all sales personnel can query the Customer, Transaction, Work and Artist tables, can change the Customer table, and can insert into the Customer and Transaction tables.



	CUSTOMER	TRANSACTION	WORK	ARTIST
Sales personnel	Insert, change, query	Insert, query	Query	Query
Management personnel	Insert, change, query	Insert, change, query	Insert, change, query	Insert, change, query
System administrator	Grant rights, modify structure	Grant rights, modify structure	Grant rights, modify structure	Grant rights, modify structure

Figure 5.1 Processing Rights at View Ridge Gallery

(Source: Kroenke, D and D. Auer. (2016). Database Processing: Fundamentals, Design and Implementation Edition 14. Pearson, Figure 9-14)

Instead of giving processing rights to specific users, processing rights are usually given to the various roles in an organisation. This makes giving and removing processing rights more manageable, as you will see in the Section 1.2.

While the DBMS can impose certain checks to ensure that operations are legitimate, e.g., inserting a previously blacklisted customer into the Customer table is illegitimate for sales personnel but legitimate for management personnel, it should be noted that the DBMS cannot ensure that all operations are legitimate, e.g., updating an employee's salary.

As such, when processing rights are given to users, processing responsibilities must be conveyed to them in some written form, so that users are aware of their responsibilities.

The rights to process data are not given to the role of database administrator, for security reasons. In addition, all operations on the database are recorded in a transaction log to ensure that every database operation is traceable to a user. A transaction log is also used during database recovery, covered in chapter 2.

Read

Kroenke, D and D. Auer. (2016). *Database Processing: Fundamentals, Design and Implementation Edition 14*. Pearson, 472-473.

Activity 1

Reproduced from Question 9.37 of the course text.

What is SQL Data Control Language (DCL)? Explain the necessity of defining processing rights and responsibilities. How are such responsibilities enforced, and what is the role of SQL DCL in enforcing them?

1.2 Database Security

1.2.1 Login

In SQL Server, a user must first have a login to access the database server. A user login allows a user to make a connection to the database engine.

To create a login in SQL Server, execute the statement using a system account (such as a database administrator):

```
create login LoginName with password = 'somePassword'
```

A login can be used to create a user for a database. Note that the same login can be used to create users for different databases. Therefore, the same login and password can be used to get a connection to access different databases.

1.2.2 User and Role

In SQL Server, databases are accessed by users created either with or without a login.

- Create user

  We create two users, *`username1`* and *`username2`* for a database `aDatabase`, *`username1`* is created with a login and *`username2`* without login. Use the statements:

  ```
  use aDatabase
  Go
  create user username1 for LoginName
  create user username2 without login
  ```

  A user without login is usually an application rule. For example, processing rights can be given to a user without a login, and a stored procedure can execute as that user, that is, execute with the processing rights of that user.

  Managing processing rights for individual user is usually tedious, especially if there are many users with the same processing rights. The processing rights must be granted to each individual user when needed. The processing rights must also be removed from each individual user when the job responsibilities are changed.

- Create a role

  To avoid having to manage processing rights at the level of individual users, a role is created. To create a role for a database `aDatabase`, use the statement:

  ```
  use aDatabase
  Go
  create role rolename
  ```

  Once a role is created, users can be added as members of the role. Members can also be removed from a role.

  - To add a user to a role

    Execute the statement:

    ```
    exec sp_addrolemember @roleName = roleName, @memberName =
    username
    ```

  - To remove a user from a role

    Execute the statement:

    ```
    exec sp_droprolemember @roleName = roleName, @memberName
    = username
    ```

Activity 2

Reproduced from Question 9.38 of the course text.

Explain the relationships among USER, GROUP, PERMISSION, and OBJECT for a generic database security system.

1.2.3 Granting, Denying and Revoking Permission

Processing rights or permissions can be given to both users and roles. Each user and each role should be given only those permissions they need, that is, they should have least possible privileges required for them to perform their job responsibilities.

Refer to Figure 5.2 for a model of database security.

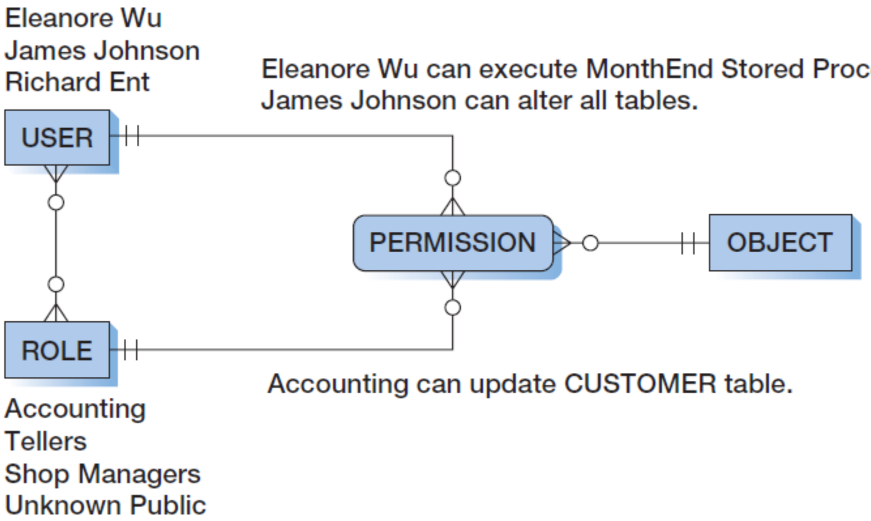

Eleanore Wu
James Johnson
Richard Ent
Eleanore Wu can execute MonthEnd Stored Procedure.
James Johnson can alter all tables.
USER
PERMISSION
OBJECT
ROLE
Accounting can update CUSTOMER table.
Accounting
Tellers
Shop Managers
Unknown Public

Figure 5.2 A Model of DBMS Security

(Source: Kroenke, D and D. Auer. (2016). Database Processing: Fundamentals, Design and Implementation Edition 14. Pearson, Figure 9-15)

SQL Data Control Language (DCL) provides 3 commands for processing rights:

- `Grant`

  ```
  grant operationsToAllowCommaSeparated on databaseObject to
  usernameOrRoleName
  ```

  The `grant` command gives processing rights or permission to a user or a role.

  Before a permission is given, the first step is to identify the database objects and the operations that the user or role should have. We then use the grant statement to give the required permissions to a user or a role.

  It is a norm to grant permissions to a role and manage permissions at role level. Members of a role have all the processing rights that the role has. In addition, using the command `deny`, specific members can be denied a processing right.

- `Revoke`

  ```
  revoke operations on databaseObject to usernameOrRoleName
  ```

  The `revoke` command removes processing rights from a user or from a role

- `Deny`

  ```
  deny operations on databaseObject to username
  ```

  The `deny` command denies processing rights from a user in a role.

Read

Kroenke, D and D. Auer. (2016). *Database Processing: Fundamentals, Design and Implementation Edition 14*. Pearson, 472-473.

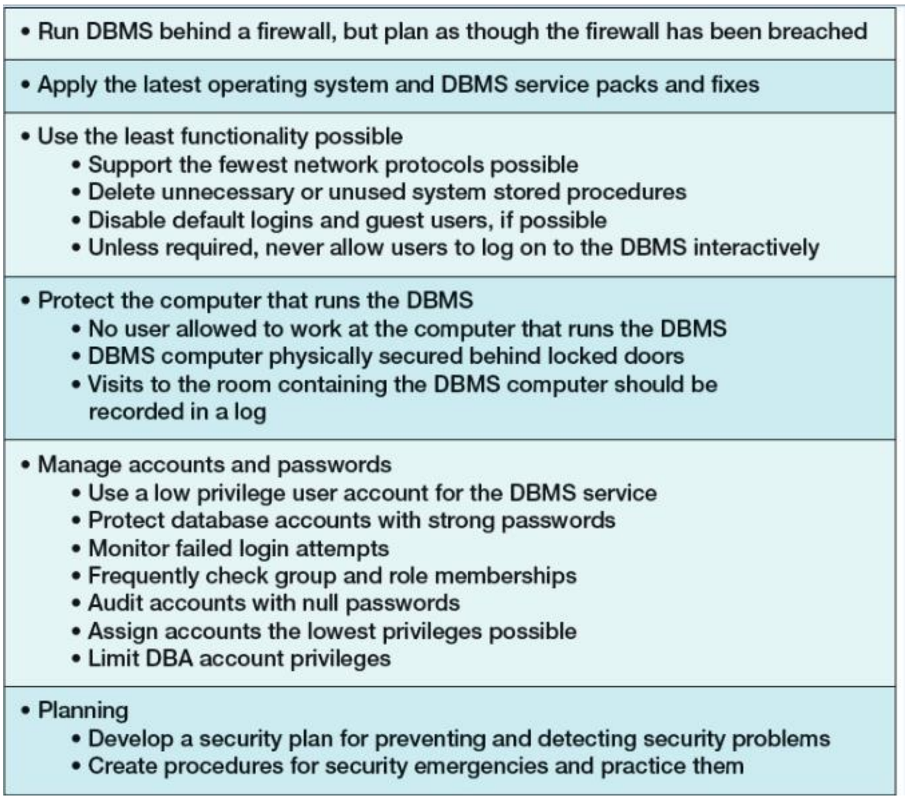
1.3 Database Security Guideline

Data is a crucial resource which an organisarion uses for its daily operations and for decision-making. Thus, it is of utmost importance to secure the database server.

The machine that hosts the database server must run behind a firewall. The database server must be on a separate machine from the web server. Furthermore, the machine must be in locked room with CCTV, with every access to the room recorded.

Updates to the DBMS must be performed regularly so that patches can fix security loopholes and new security features should be applied as soon as they are available.

A summary of database guidelines is shown in Figure 5.3.



- Run DBMS behind a firewall, but plan as though the firewall has been breached

- Apply the latest operating system and DBMS service packs and fixes

- Use the least functionality possible
  - Support the fewest network protocols possible
  - Delete unnecessary or unused system stored procedures
  - Disable default logins and guest users, if possible
  - Unless required, never allow users to log on to the DBMS interactively

- Protect the computer that runs the DBMS
  - No user allowed to work at the computer that runs the DBMS
  - DBMS computer physically secured behind locked doors
  - Visits to the room containing the DBMS computer should be recorded in a log

- Manage accounts and passwords
  - Use a low privilege user account for the DBMS service
  - Protect database accounts with strong passwords
  - Monitor failed login attempts
  - Frequently check group and role memberships
  - Audit accounts with null passwords
  - Assign accounts the lowest privileges possible
  - Limit DBA account privileges

- Planning
  - Develop a security plan for preventing and detecting security problems
  - Create procedures for security emergencies and practice them

Figure 5.3 Summary of DBMS Security Guidelines

(Source: Kroenke, D and D. Auer. (2016). Database Processing: Fundamentals, Design and Implementation Edition 14. Pearson, Figure 9-15)

Read

Kroenke, D and D. Auer. (2016). *Database Processing: Fundamentals, Design and Implementation Edition 14*. Pearson, 474-475.

Besides security measures provided by the DBMS, applications can also apply security measures, for example, those offered by the web server include secure and private data transmission. In addition, when users are allowed to make changes to SQL statements via parameters, measures must be taken to ensure that there is no possibility of SQL injection attacks.

Read

Kroenke, D and D. Auer. (2016). *Database Processing: Fundamentals, Design and Implementation Edition 14*. Pearson, 475-477.

Activity 3

Reproduced from Question 9.42 of the course text.

Why should the DBMS run on an account that has the lowest possible operating system privileges?

Activity 4

Reproduced from Question 9.43 of the course text.

What are the purposes of security planning?

Activity 5

Reproduced from Question 9.44 of the course text.

Describe the advantages and disadvantages of DBMS-provided and application-provided security.

Activity 6

Reproduced from Question 9.44 of the course text.

Describe the advantages and disadvantages of DBMS-provided and application-provided security.

Activity 6

Reproduced from Question 9.45 of the course text.

What is an SQL injection attack and how can it be prevented?

Chapter 2 Database Recovery

Recall that a database resides in secondary storage. When user requires data records, the DBMS must bring those records into memory buffers. The records are read as a whole disk page into one of the memory buffers managed by the DBMS.

The database operations requested by a user are first carried out on memory buffer. Dirtied memory buffers get written back to the secondary storage when a page is swapped out of memory buffers or when the disk pages are synchronised with the memory buffers.

Committed transactions must persist on the secondary storage. Between the time that a buffer is dirtied to the time that the buffer is written back to secondary storage, a failure may occur. The DBMS must perform database recovery to restore the database to a consistent state, as well as to ensure committed transactions are all persisted..

Note also that all database operations are logged in a transaction log file for traceability purpose as well as for recovery purpose. The transaction log is an important component of the database recovery process.

Read

Kroenke, D and D. Auer. (2016). *Database Processing: Fundamentals, Design and Implementation Edition 14*. Pearson, 477.

2.1 Recovery via Reprocessing

Recovery via reprocessing is the simplest form of recovery method. A backup of the database is periodically made. New transactions that subsequently start after the last backup are recorded. If there is a failure before another backup is made, all transactions will be reprocessed using the database backup. The transactions run anew, competing with other transactions that start after the crash.

This simplistic recovery method is not always possible if the system load is already heavy, that is, many new transactions are being started after the crash.

Reprocessing is also not possible in certain types of applications such as a booking system, e.g., a seat allocated to the reprocessed transaction may be different from that given to the transaction if it was already committed.

Reprocessing is therefore, not a choice method for database recovery.

Read

Kroenke, D and D. Auer. (2016). *Database Processing: Fundamentals, Design and Implementation Edition 14*. Pearson, 477-478.

Activity 7

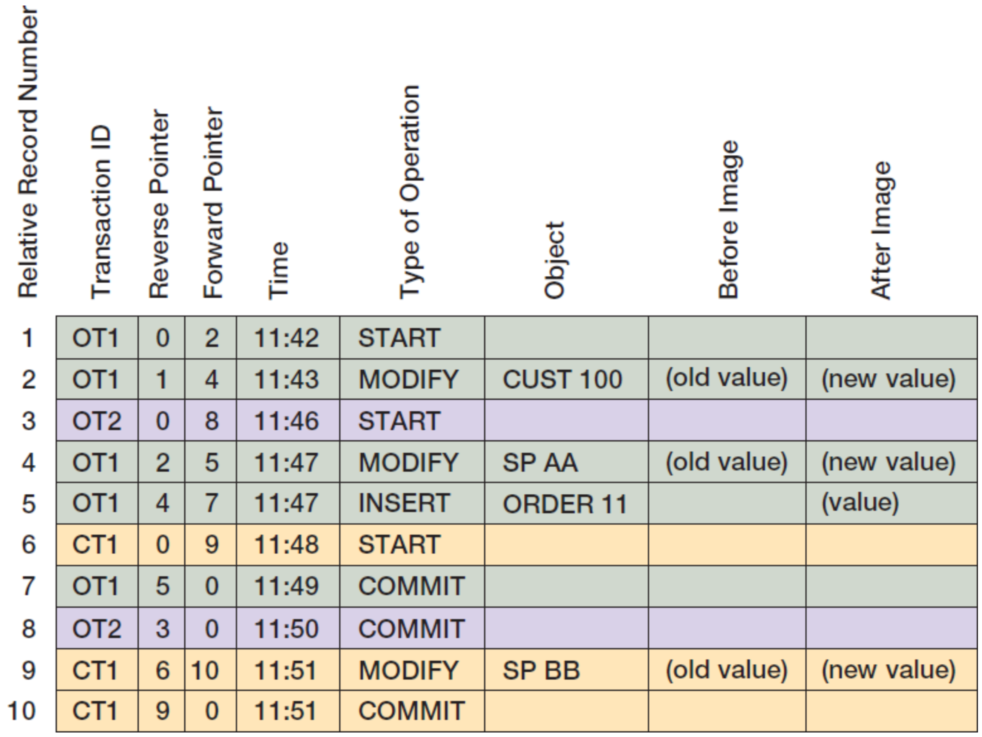
Reproduced from Question 9.46 of the course text.

Explain how a database could be recovered via reprocessing. Why is this generally not feasible?

2.2 Recovery via Rollback/Rollforward

The second database recovery method is recovery via rollback/rollforward. This recovery method requires a transaction log file. A transaction log records each database operation just before it gets executed.

Refer to Figure 5.4 for an example of what a transaction log may contain.



Relative Record Number	Transaction ID	Reverse Pointer	Forward Pointer	Time	Type of Operation	Object	Before Image	After Image
1	OT1	0	2	11:42	START			
2	OT1	1	4	11:43	MODIFY	CUST 100	(old value)	(new value)
3	OT2	0	8	11:46	START			
4	OT1	2	5	11:47	MODIFY	SP AA	(old value)	(new value)
5	OT1	4	7	11:47	INSERT	ORDER 11		(value)
6	CT1	0	9	11:48	START			
7	OT1	5	0	11:49	COMMIT			
8	OT2	3	0	11:50	COMMIT			
9	CT1	6	10	11:51	MODIFY	SP BB	(old value)	(new value)
10	CT1	9	0	11:51	COMMIT			

Figure 5.4 Summary of DBMS Security Guidelines

(Source: Kroenke, D and D. Auer. (2016). Database Processing: Fundamentals, Design and Implementation Edition 14. Pearson, Figure 9-18)

Each entry in the transaction log file describes a database operation that a transaction is about to carry out. Such logging procedure applies the write-ahead log (WAL) protocol.

Every entry in a transaction log is doubly-linked; a reverse pointer points to a database operation preceding it, and a forward pointer points to a database operation following it. The reverse pointer of the first database operation is 0, as there is no database operation preceding the first database operation. The forward pointer of the last database operation is 0, as there is no database operation following the last database operation.

Each entry records the before image or the old values of the data record before the database operation is carried out, and the after image or the new values of the data record after the database operation is carried out.

Activity 8

Reproduced from Question 9.48 of the course text.

Why is it important to write to the log before changing the database values?

In this second recovery method, the database operations of transactions that committed before a crash are not reprocessed. Instead, the before images and after images of the transaction log are used to update the database.

- Committed transactions

  The updates of committed transactions may not have been reflected in the database yet as the updates are done on the memory buffers first, and memory buffers do not get immediately written back on secondary storage.

  During the crash, the memory buffers are gone. The after images show the changes that have been made on the memory buffers. Therefore, the after image or new values of the database operations of these transaction are used to redo or rollforward specific data records. The redo starts from the beginning of committed transactions. The forward pointers help order the redo according to the same sequence as the database operations.

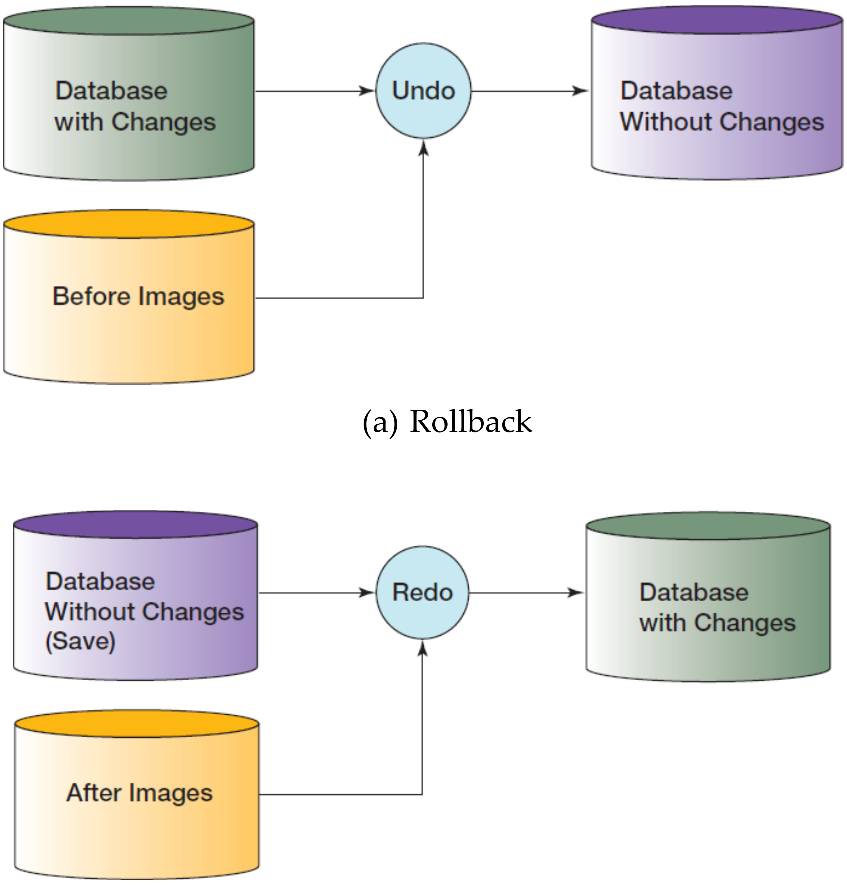
  Refer to Figure 5.5(b) for the components required for redoing a transaction.

- Rollbacked transactions

  When a transaction rollbacks, the database operations of that transaction are undone. That means the DBMS will undo the effects of the database operations of the rollbacked transaction on memory buffers first, and later on secondary storage when the memory buffers get written back on secondary storage. At the time of crash, the rollback may not reflect on the secondary storage yet.

During the crash, the memory buffers are gone. The before image or old values of the database operations of rollbacked transaction are used to undo the effects of the database operations on the database records. The undo starts from the last operation of rollbacked transactions. The reverse pointers help order the undo according to the reverse sequence of the database operations.

Refer to Figure 5.5(a) for the components required for undoing a transaction.



(a) Rollback

(b) Rollforward

Figure 5.6 Undo and Redo Transactions

(Source: Kroenke, D and D. Auer. (2056). Database Processing: Fundamentals, Design and Implementation Edition 14. Pearson, Figure 9-17)

- Incomplete transactions

When a transaction is still ongoing, some, and not all, of its database operations have been carried out. This leaves the database in an inconsistent state. As transactions are atomic, the consequence is the database operations of ongoing transactions must be undone.

The before image or old values of incomplete transactions are used to undo the database operations. The undo starts from the last operation of the ongoig transactions. The reverse pointers are used to help order the undo according to the reverse sequence of the database operations.

Refer to Figure 5.6 for an example of the recovery process for an incomplete transaction.

After the recovery process, the DBMS will restart all incomplete transactions. Incomplete transactions will compete with new transactions that start after the crash.

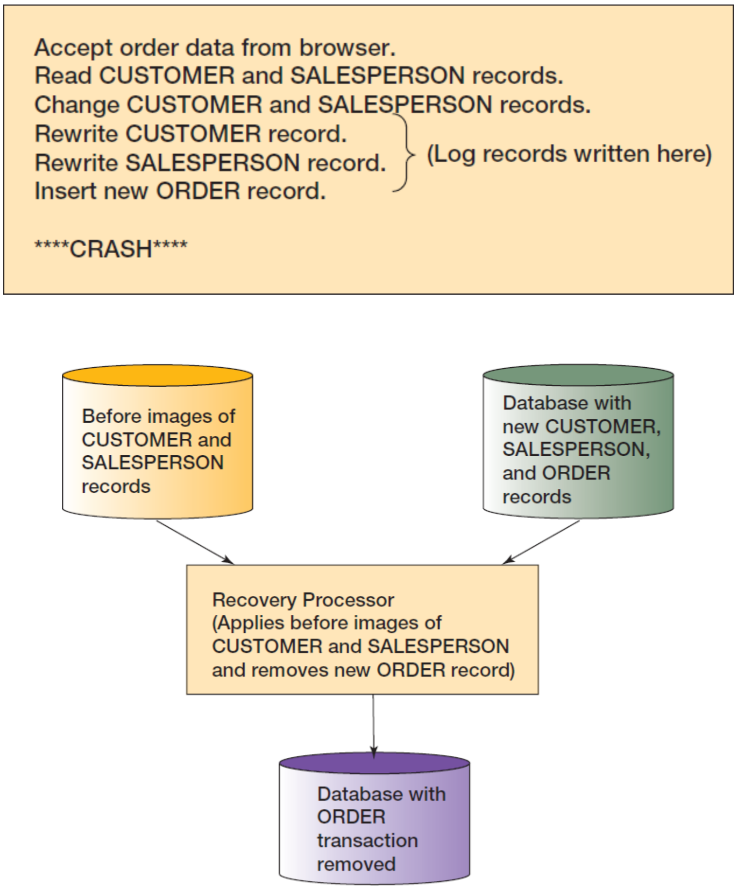

Accept order data from browser.
Read CUSTOMER and SALESPERSON records.
Change CUSTOMER and SALESPERSON records.
Rewrite CUSTOMER record.
Rewrite SALESPERSON record.
Insert new ORDER record.
(Log records written here)
****CRASH****
Before images of CUSTOMER and SALESPERSON records
Database with new CUSTOMER, SALESPERSON, and ORDER records
Recovery Processor (Applies before images of CUSTOMER and SALESPERSON and removes new ORDER record)
Database with ORDER transaction removed

Figure 5.6 Recovery Example

(Source: Kroenke, D and D. Auer. (2016). Database Processing: Fundamentals, Design and Implementation Edition 14. Pearson, Figure 9-19)

Read

Kroenke, D and D. Auer. (2016). *Database Processing: Fundamentals, Design and Implementation Edition 14*. Pearson, 478-479.

Activity 9

Reproduced from Question 9.47 of the course text.

Define rollback and rollforward.

Activity 10

Reproduced from Question 9.49 of the course text.

Describe the rollback process. Under what conditions should it be used?

Activity 11

Reproduced from Question 9.50 of the course text.

Describe the rollforward process. Under what conditions should it be used?

2.3 Checkpoints

The time a database takes to recover depends on the number of transactions to redo and undo. The more database operations there are in the transaction log file to undo and redo, the longer the database recovery time is after a crash. The redo starts from the beginning of committed transactions and the undo starts from the last operation of the rollbacked and ongoing transactions. In order to shorten the recovery time, check points are introduced.

A check point is a point of time when the DBMS will not accept new database requests so that all dirtied memory buffers can be written back to secondary storage. During this time, all the database operations in the transaction log are made to reflect on the database. Therefore, the transaction log and the database are being synchronized during a checkpoint.

Activity 12

Reproduced from Question 9.51 of the course text.

What is a checkpoint?

The effect of a checkpoint is all database operations of all transactions before the checkpoint have persisted on the secondary storage.

- Committed transactions

  All committed transactions before the last checkpoints will have their database operations persisted on the secondary storage.

- Rollbacked transactions

  All rollbacked transactions before the last checkpoints will have the effects of the rollback on database operations persisted on the secondary storage.

- Ongoing transacations

  All database operations carried out by ongoing transactions are persisted on the secondary storage, even though the transactions are incomplete, that is, have not committed or rollbacked.

With consideration to the effect of a checkpoint, the database recovery process is revised as follows:

- Committed transactions

  Nothing needs to be done for transactions that commit before the last checkpoint as their database operations have been persisted on the secondary storage during the last checkpoint.

  For transactions that commit between the last checkpoint and the crash, the recovery process performs a rollforward using the after images for each of their database operations after the last checkpoint.

- Rollbacked transactions

  Nothing needs to be done for transactions that rollbacked before the last checkpoints as the effects of the rollback on database operations have persisted on the secondary storage.

  For transactions that rollbacked between the last checkpoint and the crash , the recovery process performs an undo using the before images for each database operation in reverse order to the start of each rollbacked transaction.

- Ongoing transactions

The database operations carried out by ongoing transactions have persisted on the secondary storage, even though the transactions are incomplete. Their effects must be undone.

The recovery process performs a rollback for transactions in progress, undoing each database operation using the before images, in reverse order to the start of each ongoing transaction. These transactions must be restarted.

Read

Kroenke, D and D. Auer. (2016). *Database Processing: Fundamentals, Design and Implementation Edition 14*. Pearson, 478-479.

Summary

Security and database recovery are two important facilities that provided by a DBMS. Security is about ensuring that a user has legitimate access to perform a database operation. Database recovery is about restoring the database when there is a system crash or when the database has been corrupted.

One key idea about database security model is managing database access to ensure that users and roles have been given only those permissions for the needed operations on database objects. SQL/DCL offers three new constructs: grant, revoke and deny permissions. A database administrator can use these constructs to manage database access.

Managing processing rights for individual user can be tedious. Therefore, it is a norm to create roles and grant processing rights to roles. Users can then be added to one or more roles. Members of a role have all the processing rights that the role has..

There are also many guidelines for security that a database administrator can adopt, such as to protect the physical machine that host the database from threats interenal and external threats.

Database recovery can be done by reprocessing or other methods that use the transaction logs. Most frequently, reprocessing is not feasible as the DBMS may not be able to cope with operations requests from both new transactions and committed transactions whose changes are lost. Furthermore, reprocessing committed transactions may result in a different output from those obtained in the earlier processing, as the reprocessed transactions may run under different database states.

Methods that use transaction logs apply rollforward and rollback using the after images and before images respectively. During recovery, no new request is handled. Instead, the after images of committed transactions and before images of rollbacked transactions are copied to the database. A checkpoint reduces the database recovery time.

References

Book

Author(s)	Year	Book Title	Edition	Publisher
Kroenke, D., & Auer, D. J.	2016	*Database Processing – Fundamentals, Design, and Implementation*	14	Pearson

Quiz

1. Which statement is true about processing rights?

   a. The database applications must enforce processing rights.

   b. Processing rights should be documented and encoded into manual procedures.

   *c. Processing rights may be implemented at the DBMS level.

   d. None of the above

2. Which of the following cannot be enforced in the DBMS?

   a. Processing rights

   b. Security

   *c. Processing responsibilities

   d. Transaction isolation

3. Role A has been given the permissions to query and insert into the Customer table. Role B has been given the permissions to update and insert into the Customer table but the insert permission is subsequently revoked. Suppose user 1 is a member of both roles. What permissions does user 1 have?

   a. Query only

   b. Query and update

   *c. Query, insert and update

   d. User 1 has lost all permissions.

4. Which actions are required for recovering a database via rollforward?

a. Restore the database from the last good copy and reprocess all transactions since the last good copy.

*b. Restore the database from the last good copy and reapply all the changes made by transactions since the last good copy.

c. Undo the changes made by erroneous or partially processed transactions and restart the valid transactions that were in process at the time of the failure.

d. Recreate the database by re-entering all of the data from the beginning, and then reprocess all of the transactions.

5. Which of the following data is not contained in a transaction log?

a. Before images

b. Type of operation

c. Time of the action

*d. Permissions

.

Formative Assessment

1. Which statement is false about database security?

   a. The goal of database security is to ensure that only authorized users can perform authorized activities at authorized times.

   Incorrect. This is the goal of database security. Refer to textbook page 472.

   b. All commercial DBMS products use some version of "username and password" as part of their security features.

   Incorrect. All commercial DBMS such as Microsoft SQL Server, Oracle and MySQL use some version of "username and password" as part of their security features. Refer to textbook page 474.

   c. The security provided by the DBMS often has to be augmented by additional security features within the application program.

   Incorrect. Database security is multi-pronged. Refer to textbook pages 474-476.

   d. None of the above

   Correct! All statements are true about database security. Refer to textbook pages 472-476

2. Which action cannot be detected by the DBMS?

   a. A user makes a query on a Customer table for which he has no processing right.

   Incorrect. The processing right to make query on Customer table is enforced by the DBMS. Refer to textbook pages 472-473.

   b. The spouse of a user logs into the system to query the Customer table.

   Correct. This is a processing responsibility of user that both the DBMS and application programs cannot enforce. Refer to textbook pages 472-473.

   c. A user deletes the Customer table for which he has no processing right

Incorrect. The processing right to delete the Customer table is enforced by the DBMS. Refer to textbook pages 472-473.

d. All of the above can be detected by the DBMS.

Incorrect. Not all actions can be detected by the DBMS. Refer to textbook pages 472-473.

3. Which statement is false about database security features?

a. Users can be a member of one or more roles.

Incorrect. Users can be assigned many roles. Refer to textbook page 473.

b. A role can be assigned to only one user.

Correct! The statement is false. Many users can be assigned to the same role. Refer to textbook page 473.

c. Both users and roles can have many permissions.

Incorrect. Both users and roles can have many permissions, dependent on their job responsibilities. Refer to textbook pages 473.

d. Database objects have many permissions.

Incorrect. Database objects have many permissions, dependent on the job responsibilities of their users. Refer to textbook pages 473.

4. Which actions are required for recovering a database via reprocessing?

a. Restore the database from the last good copy and reprocess all transactions since the last good copy.

Correct! These are the actions taken for recovering a database via reprocessing. Refer to textbook page 477.

b. Restore the database from the last good copy and reapply all the changes made by transactions since the last good copy.

Incorrect. These are the actions taken for recovering a database via rollforward. Refer to textbook pages 477-480.

c. Undo the changes made by erroneous or partially processed transactions and restart the valid transactions that were in process at the time of the failure.

Incorrect. These are the actions taken for recovering a database via rollback. Refer to textbook pages 477-480

d. Recreate the database by re-entering all of the data from the beginning, and then reprocess all of the transactions.

Incorrect. The database cannot be recreated by re-entering all of the data from the beginning. There must be at least one good copy of the database for recovery via reprocessing. Refer to textbook page 477.

5. Which actions are required for recovering a database via rollback?

a. Restore the database from the last good copy and reprocess all transactions since the last good copy.

Incorrect. These are the actions taken for recovering a database via reprocessing. Refer to textbook page 477.

b. Restore the database from the last good copy and reapply all the changes made by transactions since the last good copy.

Incorrect. These are the actions taken for recovering a database via rollforward. Refer to textbook pages 477-480.

c. Undo the changes made by erroneous or partially processed transactions and restart the valid transactions that were in process at the time of the failure.

Correct! These are the actions taken for recovering a database via rollback. Refer to textbook pages 477-480

d. Recreate the database by re-entering all of the data from the beginning, and then reprocess all of the transactions.

Incorrect. The database cannot be recreated by re-entering all of the data from the beginning. There must be at least one good copy of the database for recovery. Refer to textbook pages 477-480.

Solutions or Suggested Answers

Activity 1

SQL Data Control Language (DCL) is the set of SQL statement use grant and deny database processing permissions to users.

Processing rights and responsibilities are necessary to bring order to the processing of the database, which is a shared resource. While rights can be enforced by the DBMS and application programs, responsibilities must be documented and understood by users. The upholding of responsibilities cannot be automated. It's a matter of user training and behavior.

Activity 2

A USER can be assigned to one or more GROUPs (also called ROLEs), and a GROUP can have one or more USERs. Both USERs and GROUPSs have many PERMISSIONs.

OBJECTs (used in a generic, not an OOP sense) have many PERMISSIONs assigned to them. Each PERMISSION pertains to one USER or GROUP and one OBJECT.

Activity 3

The DBMS itself should run on an account that has the lowest possible operating system privileges. In that way, if an intruder were to gain control of the DBMS, the intruder would have limited authority on that local computer or network.

Activity4

The purposes of security planning are to develop procedures for both preventing and detecting security problems Furthermore, procedures should be developed for operations to be taken in case of a security breach.

Activity 5

DBMS-provided – Advantages: Easier to implement, it will be done regardless of the source of data changes and activities, probably more consistent. Disadvantages: May not suffice for particular needs. Works best for vertical security.

Application-provided – Advantages: Can be tailored to unique requirements. Can provide horizontal security. Disadvantages: May be done poorly or inconsistently, must be programmed and maintained, may not be as robust.

Activity 6

An SQL injection attack occurs when some form of SQL is included as data when a user enters data into a form. Any time user input is used to modify an SQL statement, that input must be carefully edited to ensure that only valid input has been received and that no additional SQL syntax has been entered.

Activity 7

Reprocessing means to redo all events exactly like they were done the first time. For example, if several transactions from an ATM were lost, reprocessing would mean going back to the ATM and performing the same transactions in the same order.

First, reprocessing transactions takes the same amount of time as processing them in the first place. If the computer is heavily scheduled, the system may never catch up. Second, when transactions are processed concurrently, events are asynchronous. Slight variations in human activity, such as a user inserting a floppy disk more slowly or a user reading an electronic mail message before responding to an application prompt, may change the order of the execution of concurrent transactions.

Activity 8

If the system crashes between the time a transaction is logged and the time it is applied, at worst there is a record of an unapplied transaction. If, on the other hand, the transactions were to be applied before they were logged, it would be possible (but undesirable) to change the database but have no record of the change. If this happened, an unwary user might reenter an already completed transaction.

Activity 9

In a rollback, we undo changes made by erroneous or partially processed transactions by undoing the changes they have made in the database. We apply before images to the changed database data. Then, the valid transactions that were in process at the time of the failure are restarted.

In a rollforward, the database is restored using the saved data, and all valid transactions since the save are reapplied. We apply after images to the restored database data.

Activity 10

In a rollback, the current database and the transaction log are used. Before images of all uncommitted transactions are placed back on the database and any failed transaction is restarted. A rollback is used when a transaction fails or a system failure occurs that does not damage the active database.

Activity 11

In a rollforward, the saved copy of the database and the transaction log are used. First, the database is restored from the saved copy. Next, after-images of all committed transactions are placed back on the database and all failed transactions are restarted. A rollforward is used when a failure has occurred that renders the database unusable.

Activity 12

A checkpoint is a point of synchronization between the database and the transaction log.