Reliability of D-DBMS and recovery

UNIT-2

SUB-UNIT-2.4

Reliability

Reliability

- One of the critical functions that database systems have to ensure is correctness and availability.
- During the course of the database operations, the database system may stop running or some transactions may have to be aborted before the transactions commit.
- In those situations, atomicity and durability would be compromised if a committed transaction isn’t written to the disk or if aborted transaction is written to the disk.
- It is the role of the transaction manager is to guarantee correctness of database system - all actions in the transaction happen or none happen and if a transaction commits then its effects persist.

Other Fundamental Definition

Failure

The deviation of a system from the behavior that is described in its specification.

Erroneous state

The internal state of a system such that there exist circumstances in which further processing, by the normal algorithms of the system, will lead to a failure which is not attributed to a subsequent fault.

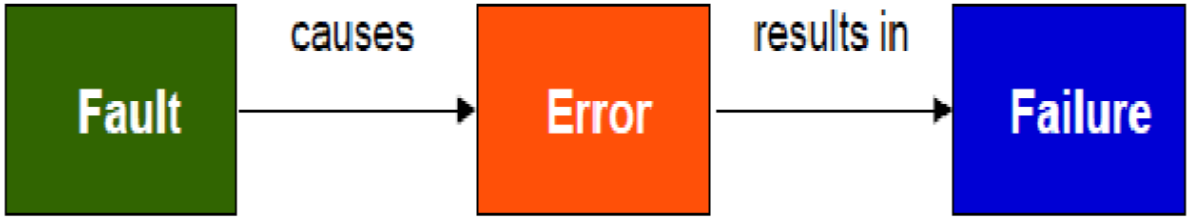
Error

The part of the state which is incorrect.

Fault

An error in the internal states of the components of a system or in the design of a system.

Failure Analysis



Fault

causes

Error

results in

Failure

Types of Faults

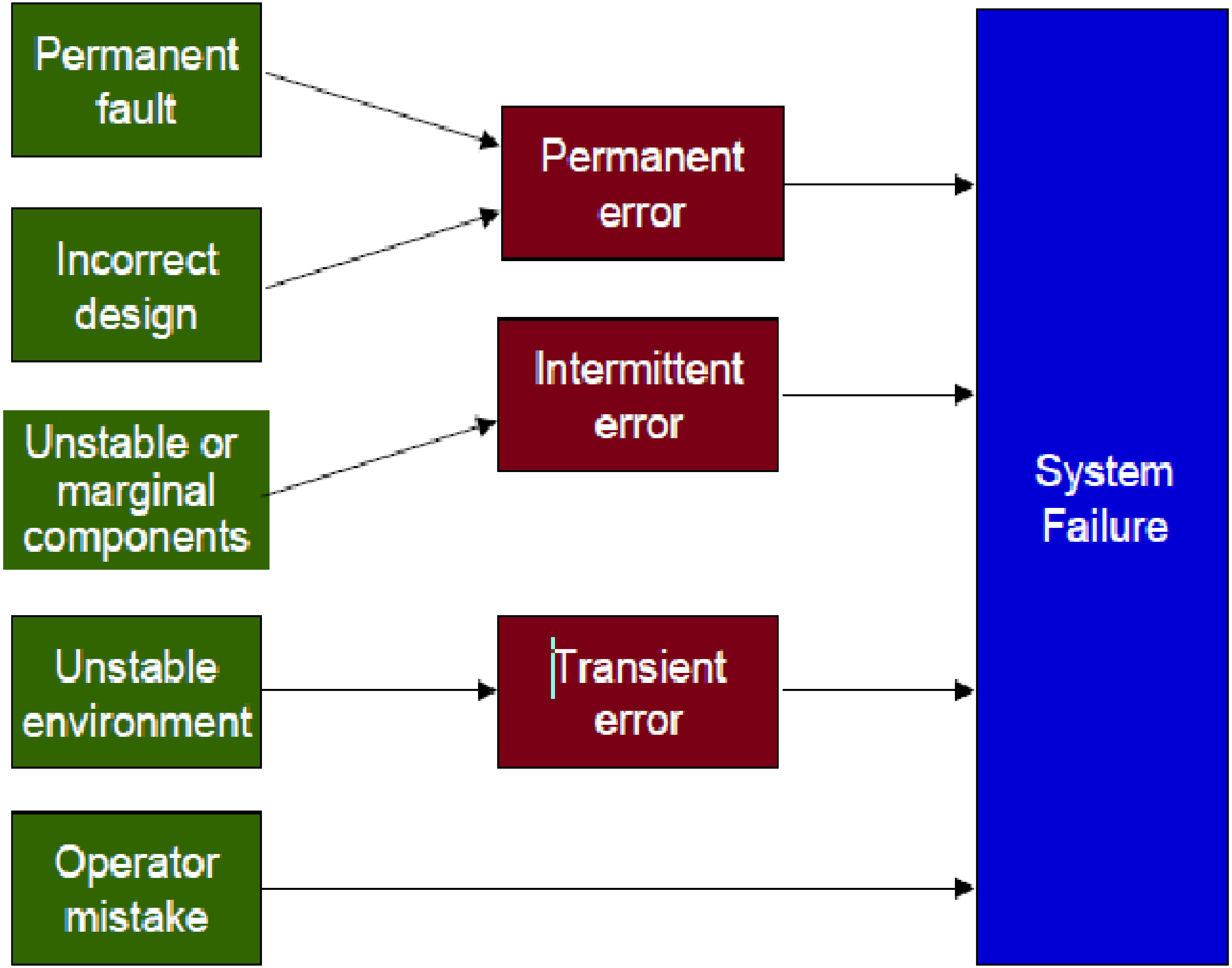
a) Hard faults

- Permanent
- Resulting failures are called hard failures.

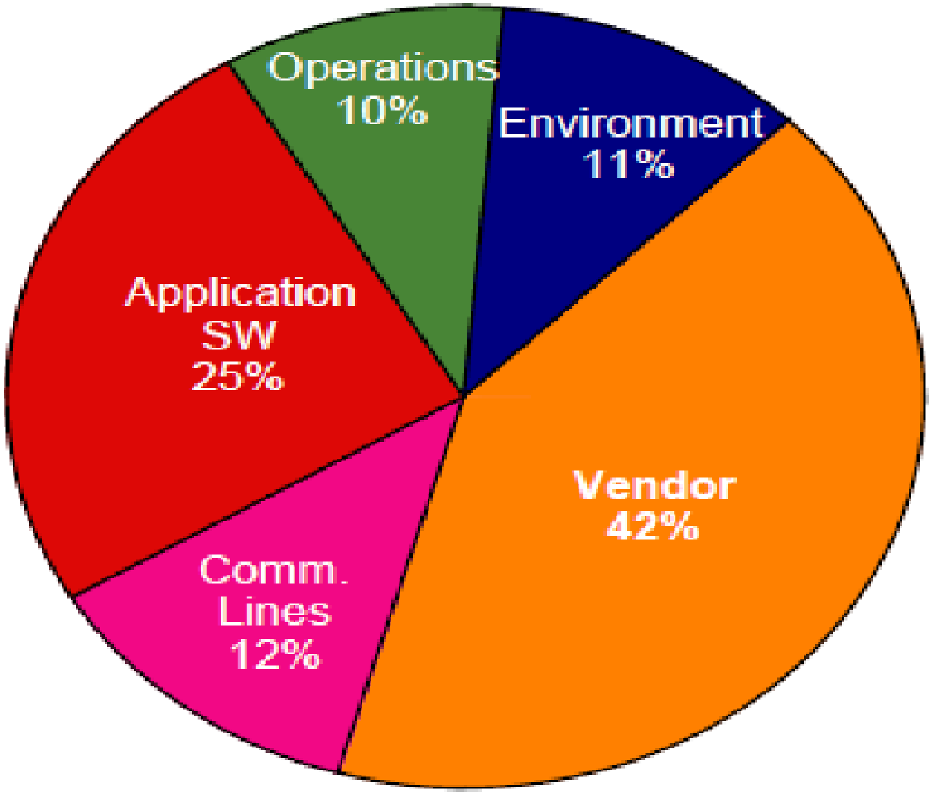
b)Soft faults

- Transient or intermittent
- Account for more than 90% of all failures
- Resulting failures are called soft failures

Faults Classification

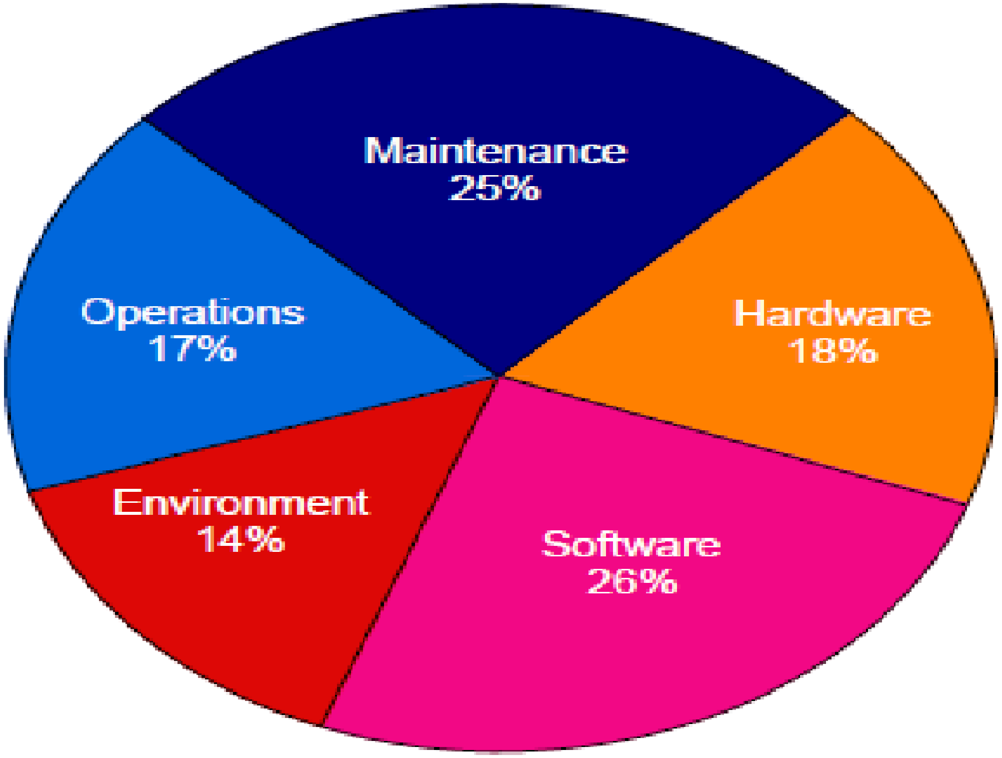


Sources of Failure Japanese Data(1986)



“Survey on Computer Security”, Japan Info. Dev. Corp.,1986.

Sources of Failure Tandem Data(1985)



Jim Gray, *Why Do Computers Stop and What can be Done About It?, Tandem Technical Report 85.7, 1985.*

Types of Failures

a)Transaction failures

- Transaction aborts (unilaterally or due to deadlock)
- Avg. 3% of transactions abort abnormally

b) System (site) failures

- Failure of processor, main memory, power supply, …
- Main memory contents are lost, but secondary storage contents are safe

Types of Failures

c) Media failures

- Failure of secondary storage devices such that the stored data is lost
- Head crash/HDD failure/controller failure (?)

d) Communication failures

- Lost/undeliverable messages
- Network partitioning

Distributed Reliability Protocol

1.) No Steal-Force Method:

- To ensure correctness, a trivial solution would be to use no steal-force method.
- In no steal-force method, the effects of transactions are held in the **buffer pool** until the transaction commits and after the transaction commits the effects are forced (written) to the disk.

Distributed Reliability Protocol

Problem:

1) In a long transaction, no steal prevents the content of the buffer pool to be replaced until the transaction commits.

2.)It leads to poor throughput in the database system as fewer items get be placed into the buffer manager and disk has to be continuously accessed.

Distributed Reliability Protocol

2.) <u>Write Ahead Logging(WAL):</u>

- To log every action and outcome of every transaction.
- For example, if the transaction decides to abort, the database system can look at the log and undo all the actions.
- If the database crashes before committed writes are written to the disk, then once the database comes back up, the recovery manager will read the logs and redo the actions of committed transactions.
- In WAL, logs are forced into a stable storage before updates take place or before a transaction commit.

Recovery Techniques

1.)Two phase Commit(2PC):

- It is one of the most popular technique for concurrency control in distributed database.
- In the basic form of 2PC, there is a **coordinator** and **subordinates** where the coordinator is the site that has initiated the transaction.
- In the **first stage**, the coordinator tries to get a uniform decision of either committing or aborting out of the subordinates .
- In the **second stage**, the coordinator relay the decision back to them.

Recovery Techniques

1.)**Two phase Commit(2PC):(contd.)**

Global Commit Rule:

- The coordinator aborts a transaction if and only if at least one participant votes to abort it.
- The coordinator commits a transaction if and only if all of the participants vote to commit it.

Recovery Techniques

Presumed abort: (2PC variation)

- When the coordinator receives an “abort” message from any of the subordinates, the coordinator will send “abort” messages to other subordinates and forget about the entire transaction, known as **presumed abort.**
- If any of the subordinates fail during the process, after the failed subordinate recover, it will ask the coordinator what has happened and from the coordinator’s log, it will find that the transaction has been aborted, and the subordinate will abort.

Recovery Techniques

Presumed commit: (2PC variation)

- When the coordinator receives a “commit” message from all the subordinates, the coordinator will send “commit” messages to other subordinates and commits transaction, known as **presumed commit.**
- If any of the subordinates fail during the process, after the failed subordinate recover, it will ask the coordinator what has happened and from the coordinator’s log, it will find that the transaction has been commited, and the subordinate will also commit.

Recovery Techniques

2.) <u>3 Phase Commit(3PC):</u>

- The coordinator will write a “prepare” log on a stable storage and will send “prepare” messages to the subordinates.
- After the subordinates receives a “prepare” message, they will decide if they can commit or not and sends the response back to the coordinator. If the subordinate decides to commit, a “ready” log will be written to a stable storage on the subordinate’s machine.
- If the coordinator receives an abort reply from any subordinates, the coordinator will send “prepare abort” message to all subordinates. If all subordinates send a ready to commit message, the coordinator will send “prepare commit” message to the subordinates and write a “global commit” log to the stable storage.

Recovery Techniques

2.) <u>3 Phase Commit(3PC):</u>

- If the subordinates get a “prepare commit” message, then the subordinates will go into “ready to commit” and send back an “okay” message back to the coordinator and appropriate log is written to the local stable storage. If the subordinates get an “prepare abort” message, the subordinates will do the same.
- When the coordinator receive “okay” messages back from all subordinates, the coordinator will send “final commit” or “final abort” message to the subordinates and writes appropriate log to a stable storage.

Recovery Techniques

The difference between 3PC and 2PC can be seen when the coordinator fails before sending “prepare commit” message. In 2PC, the subordinates will wait indefinitely until the coordinator comes back again, but in 3PC, a new coordinator is chosen and the new coordinator will perform as like.

Scalability of Replication

- Recovery protocols have to account for replication if the distributed database system supports replication.
- For the copies of the data to be consistent, the copies have to stay the same.
- This is done through updating every copy and the protocol is same as of 2PC or 3PC.
- A single copy is used as a coordinator, and the coordinator sends update commands to every copy and tries to commit the copies.