Data Security, Data Administration and Database Administration

University of California, Berkeley

School of Information

INFO 257: Database Management

Announcements

- Questions?
- Github Workshop Due This Weekend
  - Please see updated workshop
- Group Meeting/Workshop after lecture
- Remember there is a presentation the day of
  - Details are in the final project description

Lecture Outline

- Review
  - Database Administration: Security
- Database Administration: Disasters, Backup and Recovery
- Database Administration: Roles

Lecture Outline

- Database Administration: Data Integrity and Security
- Database Administration: Disasters, Backup and Recovery
- Database Administration: Roles

Transaction Control in ORACLE

- Transactions are sequences of SQL statements that ORACLE treats as a "logical unit of work"
  - From the user's point of view a private copy of the database is created for the duration of the transaction
- Transactions are started with SET TRANSACTION, followed by the SQL statements
- Any changes made by the SQL are made permanent by COMMIT
- Part or all of a transaction can be undone using ROLLBACK

Transactions in MySQL

- START TRANSACTION or BEGIN starts a transaction block (disables autocommit)
- COMMIT or ROLLBACK will commit the transaction block or return to state before the block was started
- MySQL may use different underlying database engines – the InnoDB engine also supports SAVEPOINT and ROLLBACK TO SAVEPOINT

- NOTE: This syntax can be used in any of MySQL's database engines - but it only **WORKS** when using the InnoDB engine (which can be set up when the tables are created)

Transactions in MySQL (5.0+)

- START TRANSACTION [WITH CONSISTENT SNAPSHOT] | BEGIN [WORK]
- COMMIT [WORK] [AND [NO] CHAIN] [[NO] RELEASE]
- ROLLBACK [WORK] [AND [NO] CHAIN] [[NO] RELEASE]
- SET AUTOCOMMIT = {0 | 1}

- The START TRANSACTION and BEGIN statement begin a new transaction. COMMIT commits the current transaction, making its changes permanent. ROLLBACK rolls back the current transaction, canceling its changes. The SET AUTOCOMMIT statement disables or enables the default autocommit mode for the current connection

Integrity Constraints

- The constraints we wish to impose in order to protect the database from becoming inconsistent.
- Five types
  - Required data
  - attribute domain constraints
  - entity integrity
  - referential integrity
  - enterprise constraints

Integrity Constraints

- The constraints we wish to impose in order to protect the database from becoming inconsistent.
- Five types
  - Required data
  - attribute domain constraints
  - entity integrity
  - referential integrity
  - enterprise constraints

Column Definitions in MySQL

- *column_definition*:

  data_type [NOT NULL | NULL]

  [DEFAULT *default_value*]
  [AUTO_INCREMENT]

  [UNIQUE [KEY] | [PRIMARY] KEY]

  [COMMENT '*string*']

  [COLUMN_FORMAT
  {FIXED|DYNAMIC|DEFAULT}]

  [STORAGE {DISK|MEMORY|DEFAULT}]
  [*reference_definition*]

E.g. – in MySQL

- *reference_definition*:

  REFERENCES *tbl_name* (*index_col_name*,...)
[MATCH FULL | MATCH PARTIAL | MATCH SIMPLE]

  [ON DELETE *reference_option*]

  [ON UPDATE *reference_option*]

- *reference_option*:

  RESTRICT | CASCADE | SET NULL | NO ACTION

Disasters come in many forms…

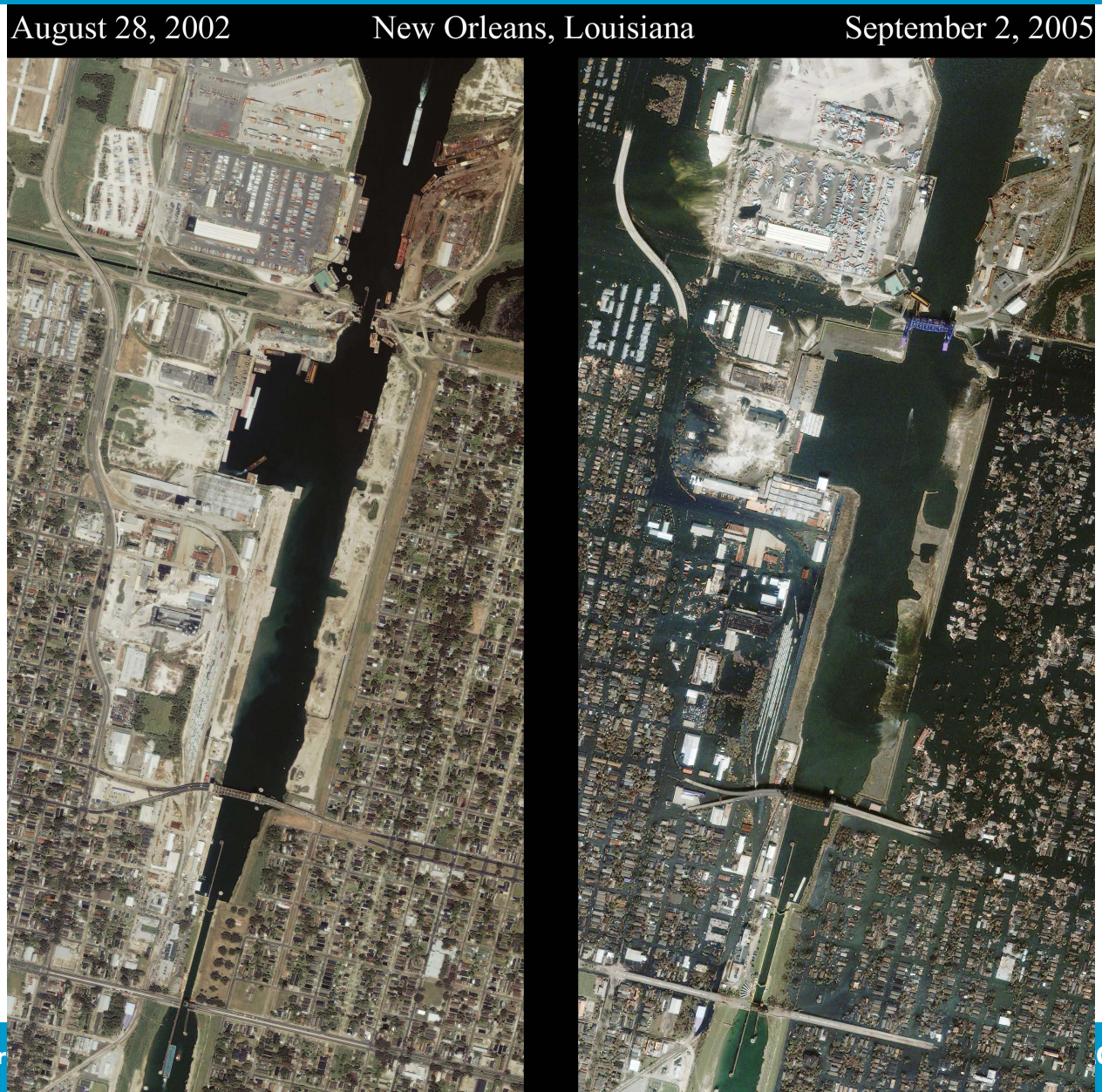
Pre Flood (Sept. 1992)

Peak Flood (Sept. 1993)

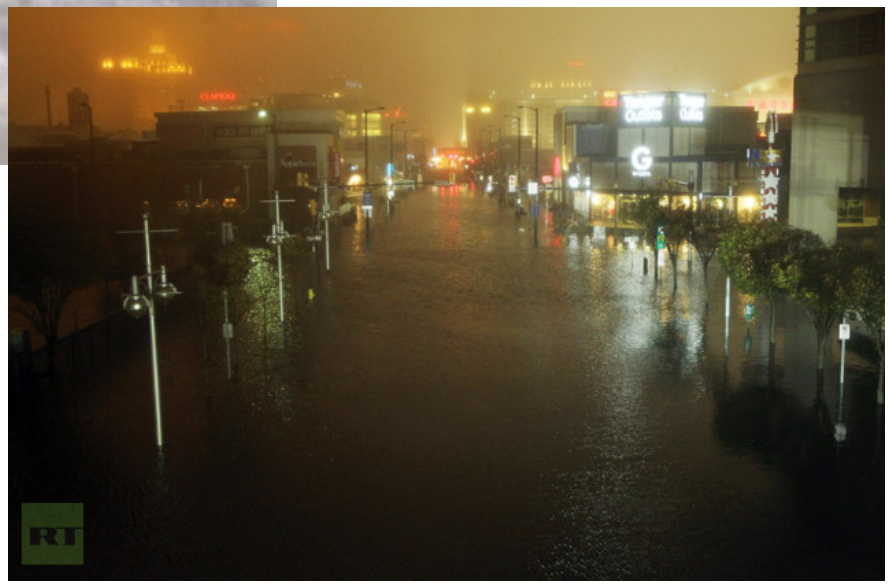
Post Flood (Oct. 1993)

La Crosse, Wisc 2001

Katrina



Hurricane Sandy in N.J. & N.Y.



Threats to Assets and Functions

- Water
- Fire
- Power Failure
- Mechanical breakdown or software failure
- Accidental or deliberate destruction of hardware or software
  - By hackers, disgruntled employees, industrial saboteurs, terrorists, or others

Threats

- Between 1967 and 1978 fire and water damage accounted for 62% of all data processing disasters in the U.S.
- The *water* damage was sometimes caused by fighting *fires*
- More recently improvements in fire suppression (e.g., Halon) for DP centers has meant that water is the primary danger to DP centers

Kinds of Records

- Class I: VITAL
  - Essential, irreplaceable or necessary to recovery
- Class II: IMPORTANT
  - Essential or important, but reproducible with difficulty or at extra expense
- Class III: USEFUL
  - Records whose loss would be inconvenient, but which are replaceable
- Class IV: NONESSENTIAL
  - Records which upon examination are found to be no longer necessary

Database Recovery

- Mechanism for restoring a database quickly and accurately after loss or damage
- Recovery facilities:
  - Backup Facilities
  - Journalizing Facilities
  - Checkpoint Facility
  - Recovery Manager

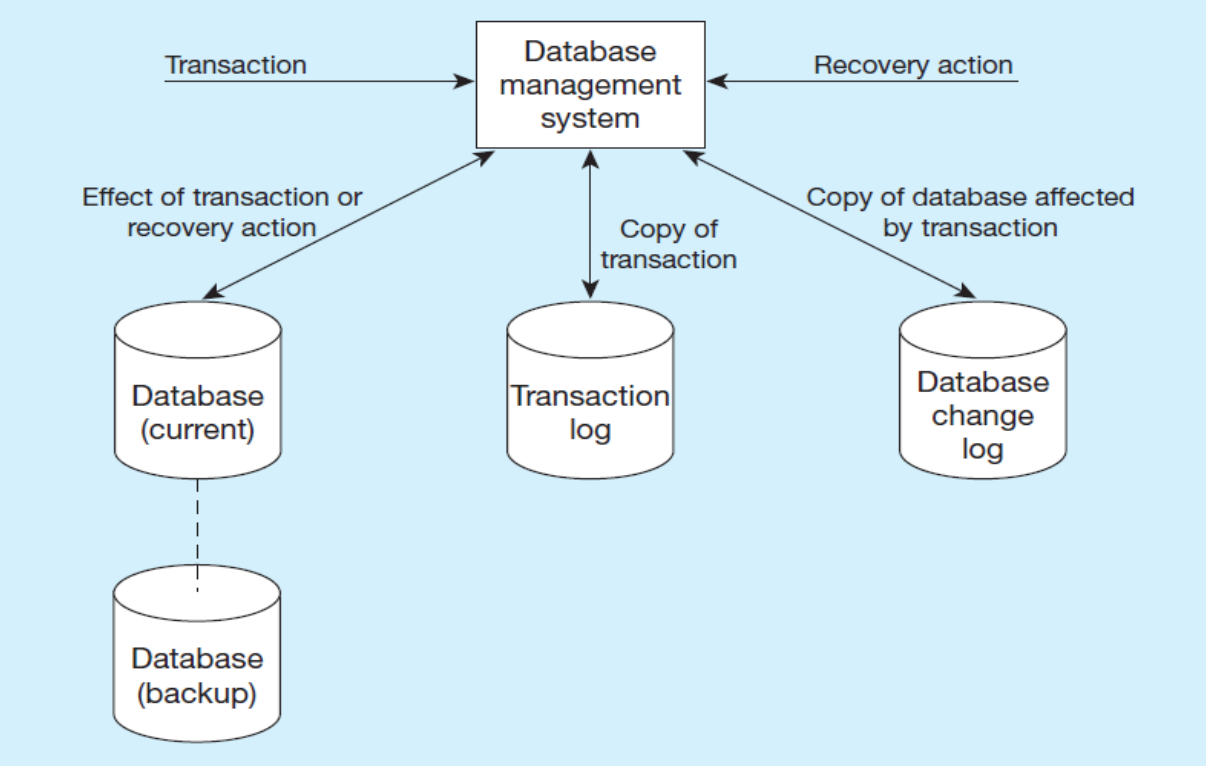
Back-up Facilities

- DBMS copy utility that produces backup copy of the entire database or subset
- Periodic backup (e.g. nightly, weekly)
- Cold backup – database is shut down during backup
- Hot backup – selected portion is shut down and backed up at a given time
- Backups stored in secure, off-site location

Journalizing Facilities

- Audit trail of transactions and database updates
- Transaction log – record of essential data for each transaction processed against the database
- Database change log – images of updated data
- Before-image – copy before modification
- After-image – copy after modification

Database Audit Trail



From the backup and logs, databases can be restored in case of damage or loss

Checkpoint Facilities

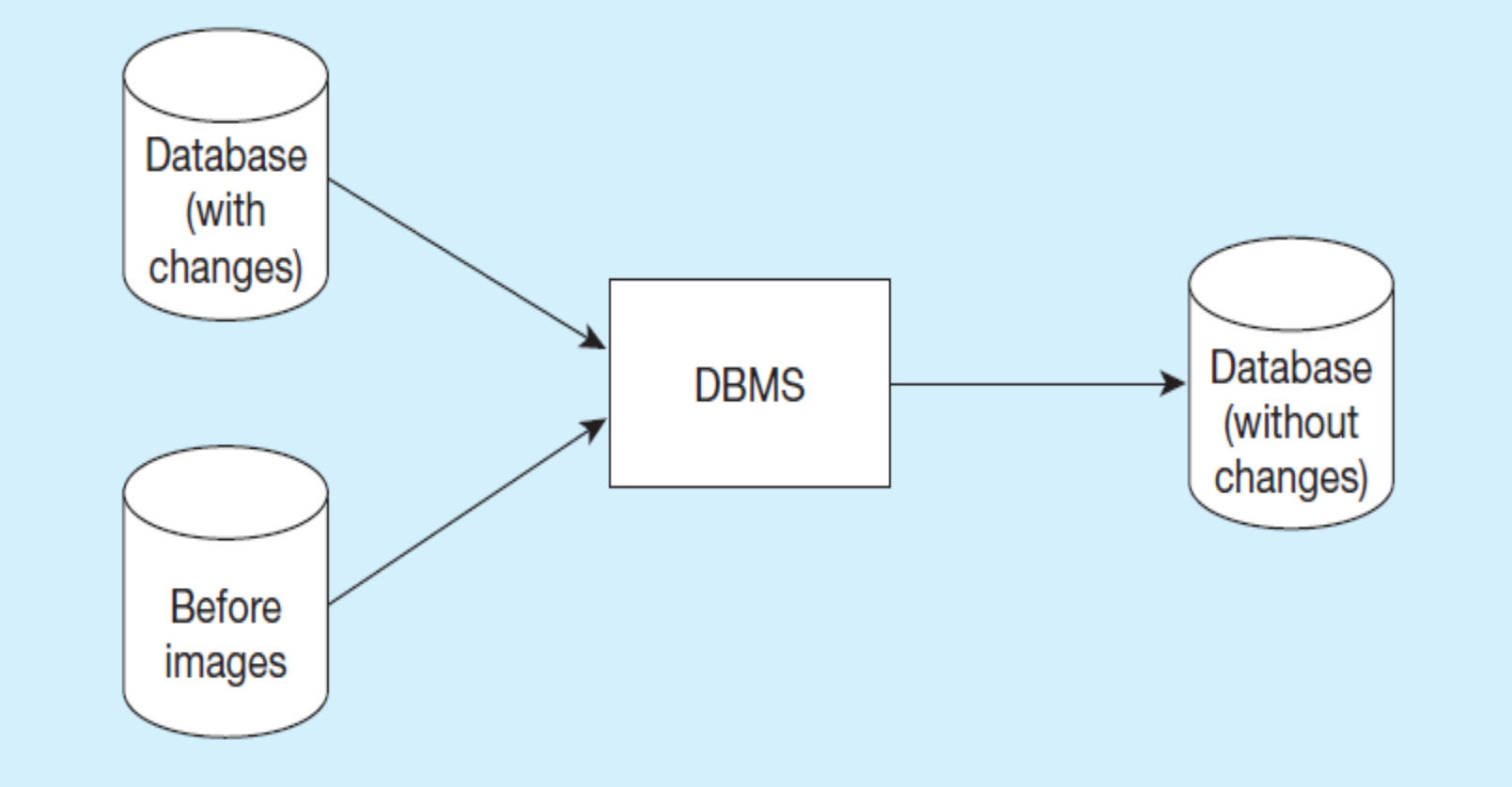
- DBMS periodically refuses to accept new transactions
- Therefore, the system is in a **quiet** state
- Database and transaction logs are synchronized
- This allows recovery manager to resume processing from short period, instead of repeating entire day

Recovery Manager

- Recovery Manager – DBMS module that restores the database to a correct condition when a failure occurs and then resumes processing user requests
- Recovery and Restart Procedures
  - Disk Mirroring – switch between identical copies of databases
  - Restore/Rerun – reprocess transactions against the backup (only done as a last resort)
  - Backward Recovery (Rollback) – apply before images
  - Forward Recovery (Roll Forward) – apply after images (preferable to restore/rerun)

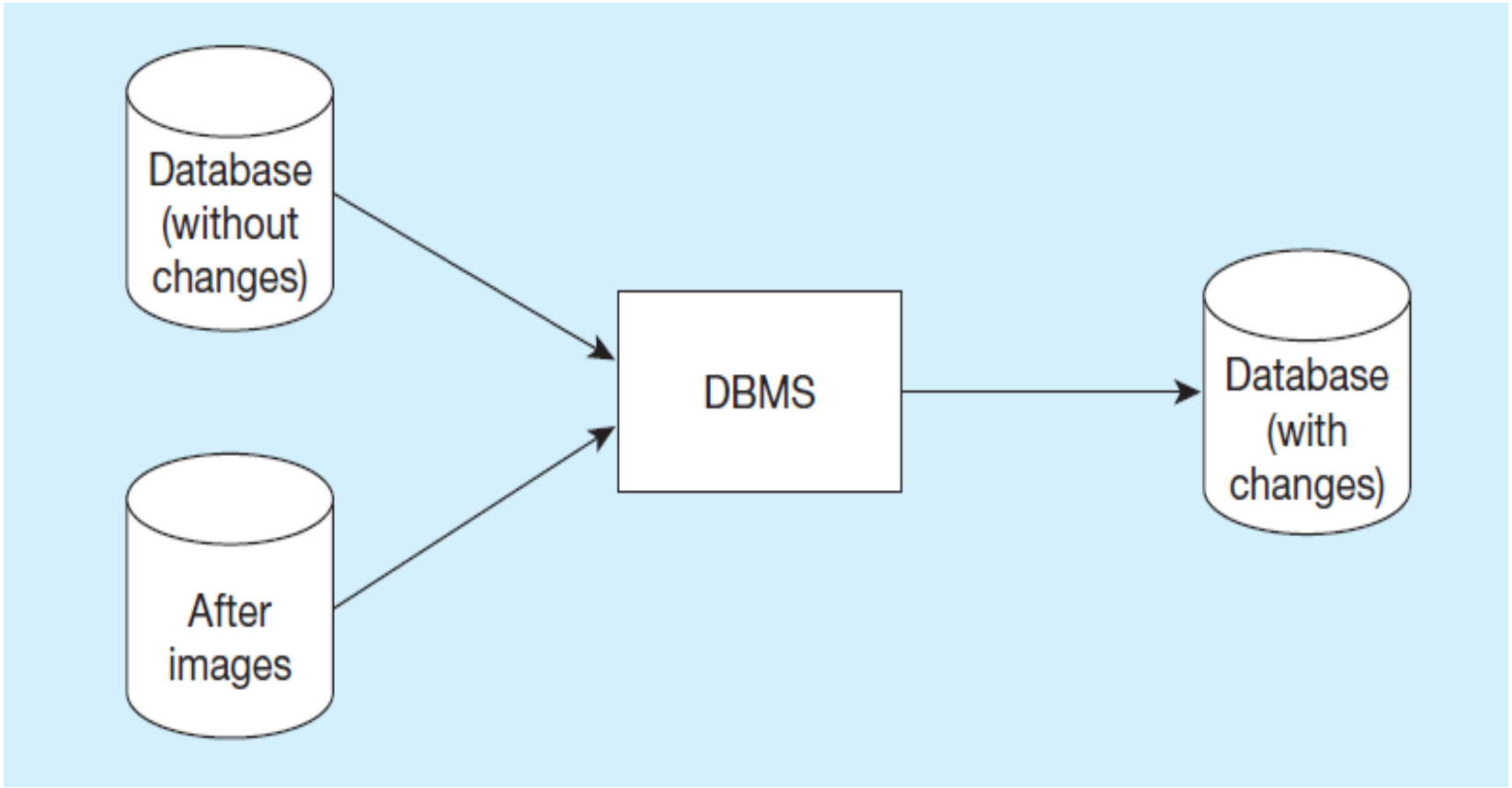
Basic Recovery Techniques (1 of 2)

a) Rollback



Basic Recovery Techniques (2 of 2)

b) Rollforward



Responses to Database Failures
(1 of 2)

- Aborted transaction
  - Rollback (preferred)
  - Rollforward/return transactions to state just prior to abort
- Incorrect data (update inaccurate)
  - Rollback (preferred)
  - Reprocess transactions without inaccurate data updates
  - Compensating transactions

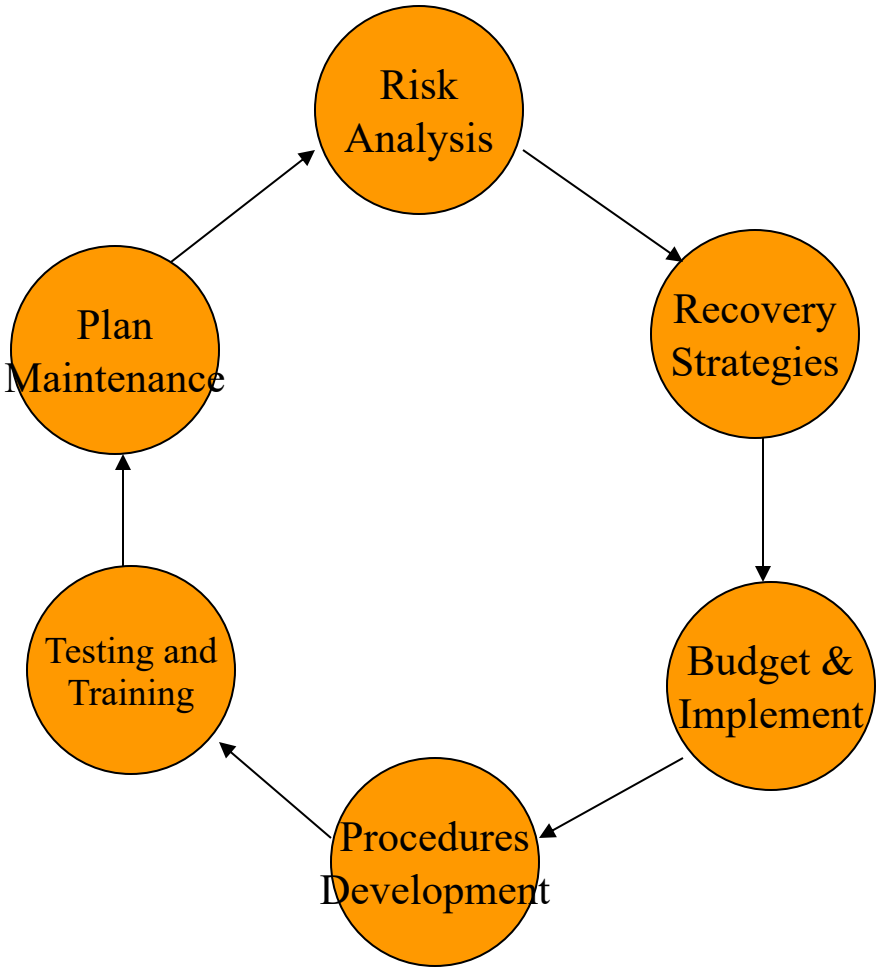
Responses to Database Failures
(2 of 2)

- System failure (database intact)
  - Switch to duplicate database (preferred)
  - Rollback
  - Restart from checkpoint (rollforward)
- Database destruction
  - Switch to duplicate database (preferred)
  - Rollforward
  - Reprocess transactions

Disaster Recovery

- Develop a detailed written disaster recovery plan, and test this regularly
- Choose and train a multidisciplinary team to carry out the plan
- Establish a backup data center at an off-site location, located a sufficient distance from the primary site
- Send backup copies of databases to the backup data center on a scheduled basis

Disaster Recovery Planning



From Toigo "Disaster Recovery Planning"

Lecture Outline

- Review
  - Database Administration: Security
- Database Administration: Disasters, Backup and Recovery
- Database Administration: Roles

Today

- Traditional and Current Data Administration
- Traditional and Current Database Administration

Changes in Traditional Roles

- This is being driven by rapid changes in
  - Technology
  - Platforms (e.g., Micro vs. Mainframe vs. Server vs. Cloud)
  - Organizational Structure
- We will focus on the core functions and tasks of these roles (traditional or current)

Traditional Administration Definitions

- ***Data Administration**:* A high-level function that is responsible for the overall management of data resources in an organization, including maintaining corporate-wide definitions and standards
- ***Database Administration**:* A technical function that is responsible for physical database design and for dealing with technical issues such as security enforcement, database performance, and backup and recovery

Traditional Data Administration Functions

- Data policies, procedures, standards
- Planning
- Data conflict (ownership) resolution
- Managing the information repository for:
  - Data definitions, business rules, and data relationships
  - Automated data modeling and design tools
  - Applications that access and manipulate data
  - Database management systems
- Internal marketing of DA concepts

Database Administration Functions

- Analyzing and designing databases
- Selecting DBMS and software tools
- Installing/upgrading DBMS
- Tuning database performance
- Improving query processing performance
- Managing data security, privacy, and integrity
- Data backup and recovery

Evolving Approaches to Data Administration

- Blend data and database administration into one role
- Fast-track development – monitoring development process (analysis, design, implementation, maintenance)
- Procedural DBAs–managing quality of triggers and stored procedures
- eDBA–managing Internet-enabled database applications
- PDA DBA–data synchronization and personal database management
- Data warehouse administration

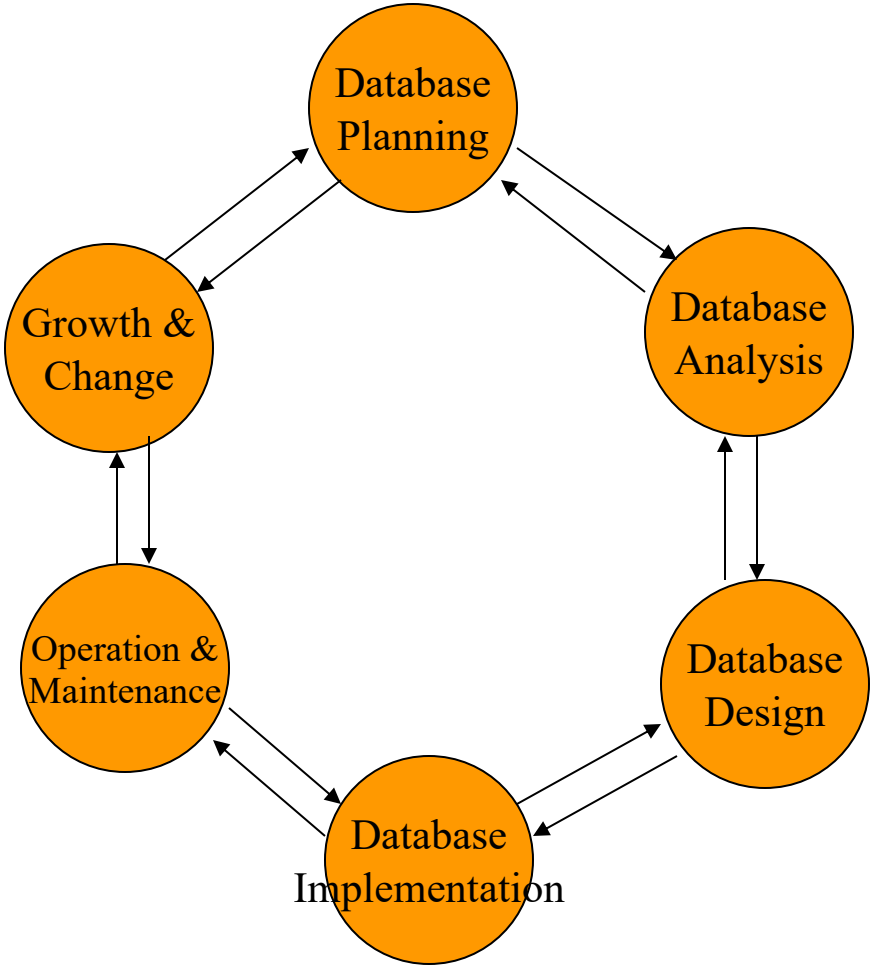
Trends in Database Administration

- Increased use of procedural logic
- Proliferation of Internet-based applications
- Increased use of mobile smart devices
- Cloud computing and database/data administration

Ineffective Data Administration leads to Poor Data Quality

- Multiple definitions, inconsistent representation of the same data entity/elements
- Missing key data elements
- Low data quality levels due to inappropriate sources of data or timing of data transfers
- Inadequate familiarity with existing data
- Poor and inconsistent query response time, excessive database downtime, and either stringent or inadequate controls
- Lack of access to data due to damaged, sabotaged, or stolen files or due to hardware failures
- Embarrassment to the organization

Database System Life Cycle



Note: this is a different version of this life cycle than discussed previously

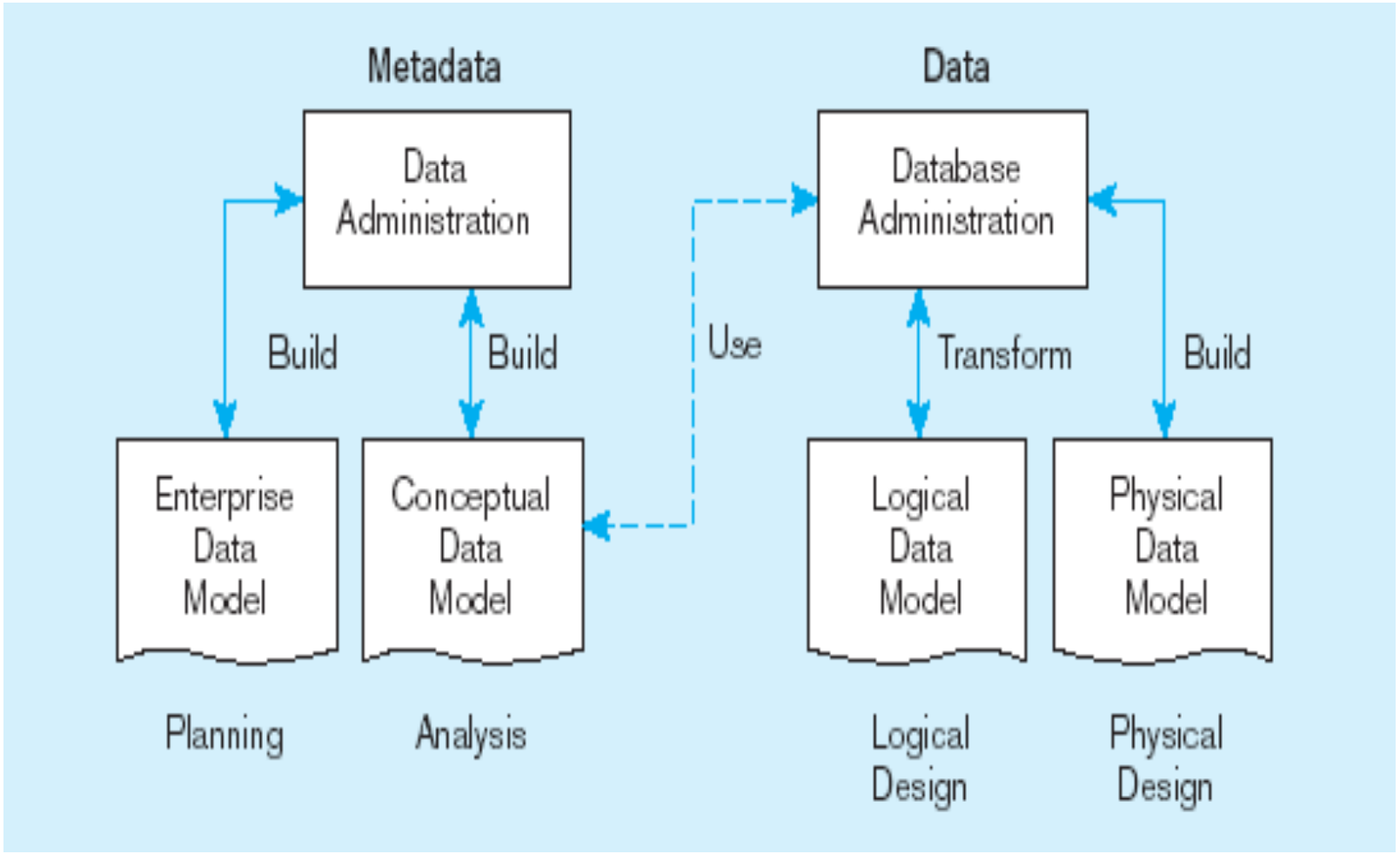
Database Planning

- Development of a strategic plan for database development that supports the overall organization's business plan
- DA supports top management in development of this plan
- The result of this stage is an *enterprise data model*

Database Design

- Purpose of the design phase is the development of the logical database design that will serve the needs of the organization and the physical design implementing the logical design
- In relational systems the outcome is normalized relations, and the data definition for a particular database systems (including indexes, etc.)

Roles for design process



Database Implementation

- Database design gives you an empty database
- Load data into the database structure
- Convert existing data sets and applications to use the new database
  - May need programs, conversion utilities to convert old data to new formats.
- Outcome is the actual database with its data

Database Implementation DA & DBA functions

- Specify database access policies (DA & DBA)
- Establish Security controls (DBA)
- Supervise Database loading (DBA)
- Specify test procedures (DBA)
- Develop application programming standards (DBA)
- Establish procedures for backup and recovery (DBA)
- Conduct User training (DA & DBA)

Operation and Maintenance 1: Operations

- Users are responsible for updating the database, DA and DBA are responsible for developing procedures that ensure the integrity and security of the database during the update process.
- Specific responsibility for data collection, editing and verification must be assigned
- Quality assurance must be practiced to protect and audit the database quality.

Operation and Maintenance 2: Maintenance

- The ongoing process of updating the database to keep it current
  - adding new records
  - deleting obsolete records
  - changing data values in particular records
  - modifying relation structures (e.g. adding new fields)
- Privacy, security, access control must be in place.
- Recovery and Backup procedures must be established and used

Operation and Maintenance: DA & DBA functions

- Monitor database performance (DBA)
- Tune and reorganize databases (DBA)
- Enforce standards and procedures (DBA)
- Support users (DA & DBA)

Data Warehouse Administration

- New role, coming with the growth in data warehouses
- Similar to DA/DBA roles
- Emphasis on integration and coordination of metadata/data across many data sources
- Specific roles:
  - Support DSS applications
  - Manage data warehouse growth
  - Establish service level agreements regarding data warehouses and data marts

Growth & Change

- Change is a way of life
  - Applications, data requirements, reports, etc. will all change as new needs and requirements are found
  - The Database and applications and will need to be modified to meet the needs of changes to the organization and the environment
  - Database performance should be monitored to maintain a high level of system performance

Database Performance Tuning

- DBMS Installation
  - Setting installation parameters
- Memory Usage
  - Set cache levels
  - Choose background processes
- Input/Output (I/O) Contention
  - Use striping
  - Distribution of heavily accessed files
- CPU Usage
  - Monitor CPU load
- Application tuning
  - Modification of SQL code in applications

Data Availability

- Downtime is expensive
- How to ensure availability
  - Hardware failures–provide redundancy for fault tolerance
  - Loss of data–database mirroring
  - Maintenance downtime–automated and nondisruptive maintenance utilities
  - Network problems–careful traffic monitoring, firewalls, and routers

Open Source DB Management (1 of 2)

- Open Source DBMSs: alternative to proprietary packages
  - Examples: MySQL, PostgreSQL
- Advantages:
  - Pool of volunteer developers and testers
  - Less expensive than proprietary packages
  - Source code available for modification
- Disadvantages
  - Absence of complete documentation
  - Ambiguous licensing concerns
  - Not as feature-rich as proprietary DBMSs
  - Vendors may not have certification programs

Open Source DB Management (2 of 2)

- Considerations when selecting an open source DBMS
  - Features
  - Support
  - Ease of use
  - Stability
  - Speed
  - Training
  - Licensing

Data Governance

- Data governance
  - High-level organizational groups and processes overseeing data stewardship across the organization
- Data steward
  - A person responsible for ensuring that organizational applications properly support the organization's data quality goals

Requirements for Data Governance

- Sponsorship from both senior management and business units
- A data steward manager to support, train, and coordinate data stewards
- Data stewards for different business units, subjects, and/or source systems
- A governance committee to provide data management guidelines and standards

Importance of Data Quality

- If the data are bad, the business fails. Period.
  - GIGO – garbage in, garbage out
  - Sarbanes-Oxley (SOX) compliance by law sets data and metadata quality standards
- Purposes of data quality
  - Minimize IT project risk
  - Make timely business decisions
  - Ensure regulatory compliance
  - Expand customer base

Characteristics of Quality Data (1 of 2)

- Uniqueness
  - Each entity exists only once within the database
- Accuracy
  - Data correctly represents the real-life objects it models
- Consistency
  - Values for data in one data set agree with the values for related data in another data set
- Completeness
  - Data having assigned values if they need to have values

Characteristics of Quality Data (2 of 2)

- Timeliness
  - Data is available when it is needed without excessive delays
- Currency
  - Data is recent enough to be useful
- Conformance
  - Data is stored, exchanged, or presented in a format as specified by their metadata
- Referential integrity
  - Data that refer to other data are unique and satisfy requirements to exist

Causes of Deteriorated Data Quality

- External data sources
  - Lack of control over data quality
- Redundant data storage and inconsistent metadata
  - Proliferation of databases with uncontrolled redundancy and metadata
- Data entry
  - Poor data capture controls
- Lack of organizational commitment
  - Not recognizing poor data quality as an organizational issue

Steps in Data Quality Improvement

- Get business buy-in
- Perform data quality audit
- Establish data stewardship program
- Improve data capture processes
- Apply modern data management principles and technology
- Apply total quality management (TQM) practices

Business Buy-in

- Executive sponsorship
- Building a business case
- Prove a return on investment (ROI)
- Avoidance of cost
- Avoidance of opportunity loss

Data Quality Audit

- Statistically profile all data files
- Document the set of values for all fields
- Analyze data patterns (distribution, outliers, frequencies)
- Verify whether controls and business rules are enforced
- Use specialized data profiling tools

Data Stewardship Program

- Roles:
  - Oversight of data stewardship program
  - Manage data subject area
  - Oversee data definitions
  - Oversee production of data
  - Oversee use of data
- Report to: business unit vs. IT organization?
- Chief data officer
  - Executive level position accountable for all data-related activities in the enterprise

Improving Data Capture Processes

- Automate data entry as much as possible
- Manual data entry should be selected from preset options
- Use trained operators when possible
- Follow good user interface design principles
- Immediate data validation for entered data

Apply Modern Data Management Principles and Technology

- Software tools for analyzing and correcting data quality problems:
  - Pattern matching
  - Fuzzy logic
  - Expert systems
- Sound data modeling and database design

Data Availability

- Cost of downtime by business type
  - Financial services/retail brokerage – $6.45 million
  - Financial services/credit authorization – $2.6 million
  - Retail/catalog sales center – $90,000
  - Travel/reservation centers – $89,500
  - Logistics/shipping services – $28,250
  - Based on Mullins (2012, p. **273**).

Table 12-4 Cost of Downtime by Availability

Availability	Minutes	Hours	Cost per Year
99.999%	5	.08	$8,000
99.99%	53	.88	$88,000
99.9%	523	8.77	$877,000
99.5%	2,628	43.8	$4,380,000
99%	5,256	87.6	$8,760,000

Based on Mullins (2012, p. **273**).

Measures to Ensure Availability

- Hardware failures – provide redundancy for fault tolerance
- Loss of data – database mirroring
- Human error – standard operating procedures, training, documentation
- Maintenance downtime – automated and non-disruptive maintenance utilities
- Network problems – careful traffic monitoring, firewalls, and routers