Database Applications

- DB + application programs running on a DBMS
- Purpose: to satisfy end-user needs
  - Traditional applications: business data processing
  - Advanced business applications
    - Decision support
    - Workflow (office automation)
  - Nontraditional applications
    - Design (buildings, bridges, telecommunications, software)
    - AI (knowledge representation, autonomous agents, data mining)
    - Multimedia (video clips, spatial data, content-based image retrieval)
    - WWW (search engines, digital libraries, electronic commerce)
    - Business Reengineering (change management, workflow evolution)

Inadequacy of Standard Programming Languages

- Building advanced applications requires too much work.
- Standard programming languages lack:
  - Persistence
  - Data modeling
  - DB system features
    - Crash recovery
    - Integrity checking
    - Security
    - Concurrency control
    - Query optimization
    - File organization and optimization

Inadequacy of Traditional Database Systems

- Building advanced applications requires too much work.
- Problems with traditional DBs
  - Tuned for data processing
  - Simple records
  - Short-duration transactions
  - Lack support for
    - Complex data (large objects, multimedia)
    - Behavior modeling (complex operations, active behavior)
    - Long-running interactions
    - Time-based data and version control

Object Technology

- Unifying paradigm
  - Models both data and behavior.
  - Integrates programming languages and databases.
- Provides a direct real-world view
  - 1-1 correspondence between actual and represented objects.
  - Simulation of real-world object behavior.
- Wide range of characteristics.
  - Simple / complex
  - Passive / reactive / proactive
  - Lacking intelligence / exhibiting intelligence
  - Stable / changeable / adaptable

Database Application Development

- Model objects – their relationships, behavior, and interactions.
- Transform modeled objects into an implementation.
  - Approach
    - Systematic, rigorous, model-driven
    - Tool support based on concepts and principles
  - Requirements
    - Knowledgeable developer
    - A model that is formal, seamless, language-equivalent, and ontological