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