Thinking about Features

- Can use the system features to refine design
  - Identify class responsibilities - what is necessary to implement features
  - Identify class relationships - how should classes interact
  - Eliminate classes if they are not used

Feature: Customer Places Order

- New customer scenario:
  - Give identification, shipping, and billing info
  - Specify order
    - for each bicycle in order specify
      - frame
      - components
    - Commit to order

- Existing Customer scenario
  - “Login” with customer id
  - Specify order
    - for each bicycle in order specify
      - frame
      - components
    - Commit to order

Objects used in Placing Order

- Order
- Bicycle = frame + components
- Customer
- Replacements
- Inventory
- Order DB
- Customer DB

Current Class Relationships
Inventory Relationships

Order Relationships Again

Design Issue

Example Implicit Relationship

Putting Bike into Production
Handling Replacements

Approach 1: replacements found before passing order

Order DB → check availability → Inventory

Find replacements → Replacements

Handling Replacements (2)

Approach 2: replacements found during production

Production UT → check availability → Inventory

Find replacements → Replacements