Problem 5.1: Component-Based vs. Service-Oriented Architecture

- Component-Based Architecture (CBA): Focuses on reusable software components that are often part of the same application. These components interact with each other through well-defined interfaces and are typically deployed together.
- Service-Oriented Architecture (SOA): Involves loosely coupled services that communicate over a network. Services are independent and can be used by different applications.

Key Difference:
CBA is more about modularity within an application, while SOA focuses on interoperability across applications using distributed services.

Problem 5.2: Architecture for a Tic-Tac-Toe App

- Since the app runs entirely on a phone and doesn't use an external database, a component-based architecture is best.
- The AI opponent can be implemented as a local module.
- High scores are stored locally.
- An Event-Driven architecture would work well.

Problem 5.4: Architecture for an Online Chess Game

- Unlike Tic-Tac-Toe, this requires networked communication, so client-server architecture is ideal.
- The chess game can use a service-oriented approach if it integrates with external ranking databases or chess engines.
- The game logic can be component-based but with extra web socketing .

Problem 5.6: Database for ClassyDraw

- Database Structure:
    - Store drawings as records
    - Use a relational database if structure and relationships are important .
    - Save vector-based drawings
- Maintenance:
    - Regular indexing for performance.
    - Backup strategy to prevent data loss.
    - Possible cloud sync or local database caching.

Problem 5.8: State Machine Diagram for Scientific Notation Parsing

- The state machine should handle:
    1. Start (waiting for input)

2. Sign (`+` or `-`)
3. Integer part (digits before the decimal)
4. Decimal part (`.` followed by digits)
5. Exponent marker (`E` or `e`)
6. Exponent sign (`+` or `-`)
7. Exponent digits (digits after `E/e`)
8. End state (valid number)

- It should reject invalid input sequences.

Problem 6.1: Properties of ClassyDraw Shapes

Shared Properties (common to all shapes):

- Position (x, y)
- Size (width, height)
- Color
- Line thickness
- Rotation
- Visibility

Non-Shared Properties (specific to certain shapes):

- Text: Font, text content
- Star: Number of points
- Ellipse: Radius (instead of width/height)
- Rectangle: Corner radius (if supporting rounded rectangles)

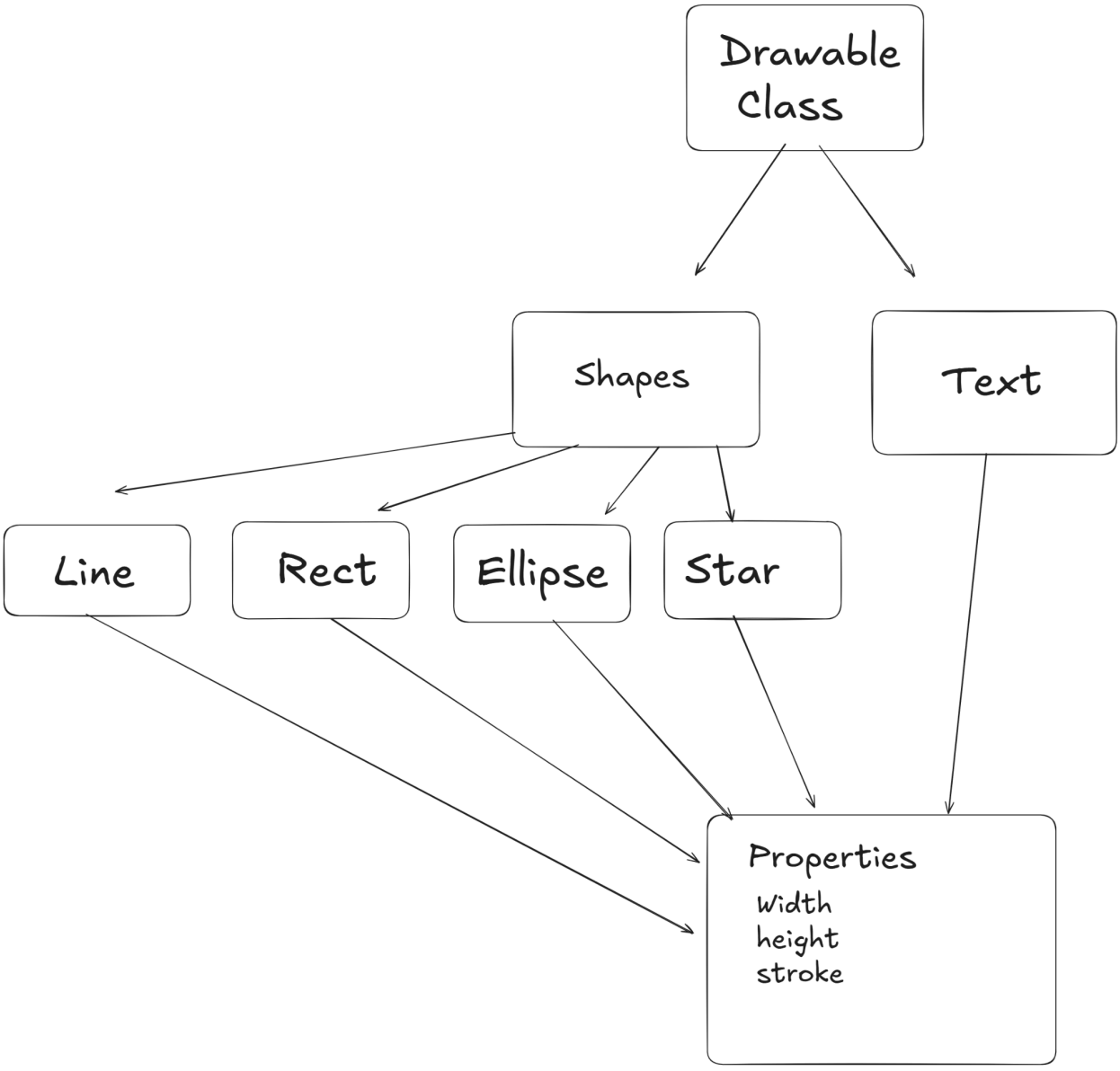
Shared by Some, Not All:

- Line, Rectangle, Ellipse, Star share stroke properties
- Text doesn't need stroke width but needs font properties.

Where to Implement:

- A base Drawable class should hold shared properties.
- Subclasses override and extend as needed.

Problem 6.2: Inheritance Diagram


Drawable Class
Shapes
Text
Line
Rect
Ellipse
Star
Properties
Width
height
stroke