Inheritance

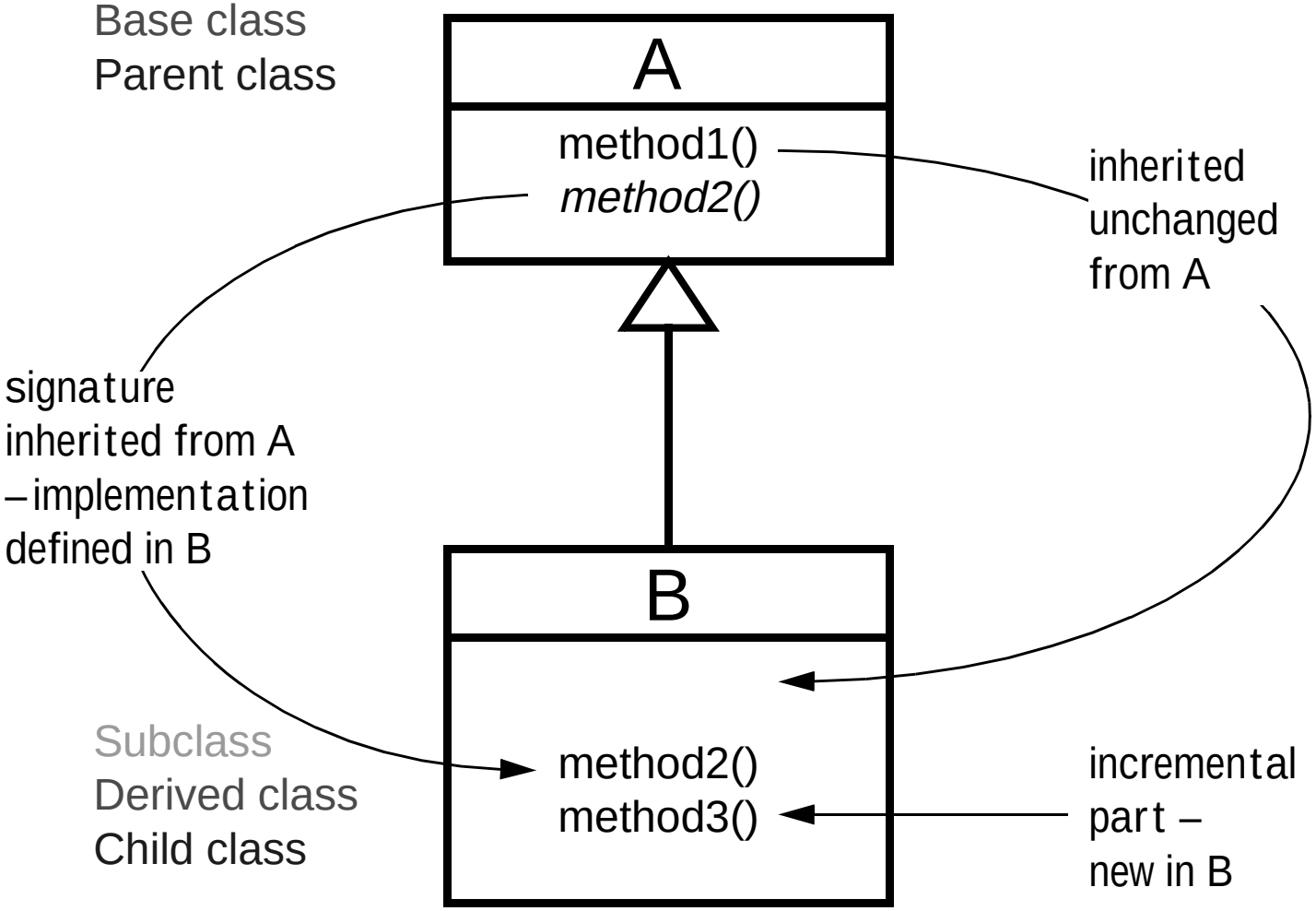
Define

- Inheritance is a generalization/specialization relationship between classes (i.e., between definitions). Using inheritance, the specification and implementation of one class is defined in terms of another class.
- Inheritance creates a hierarchy of related classes that supports an incremental style of development in which new classes reuse and extend previously created classes.

Inheritance

Document

Superclass
Base class
Parent class



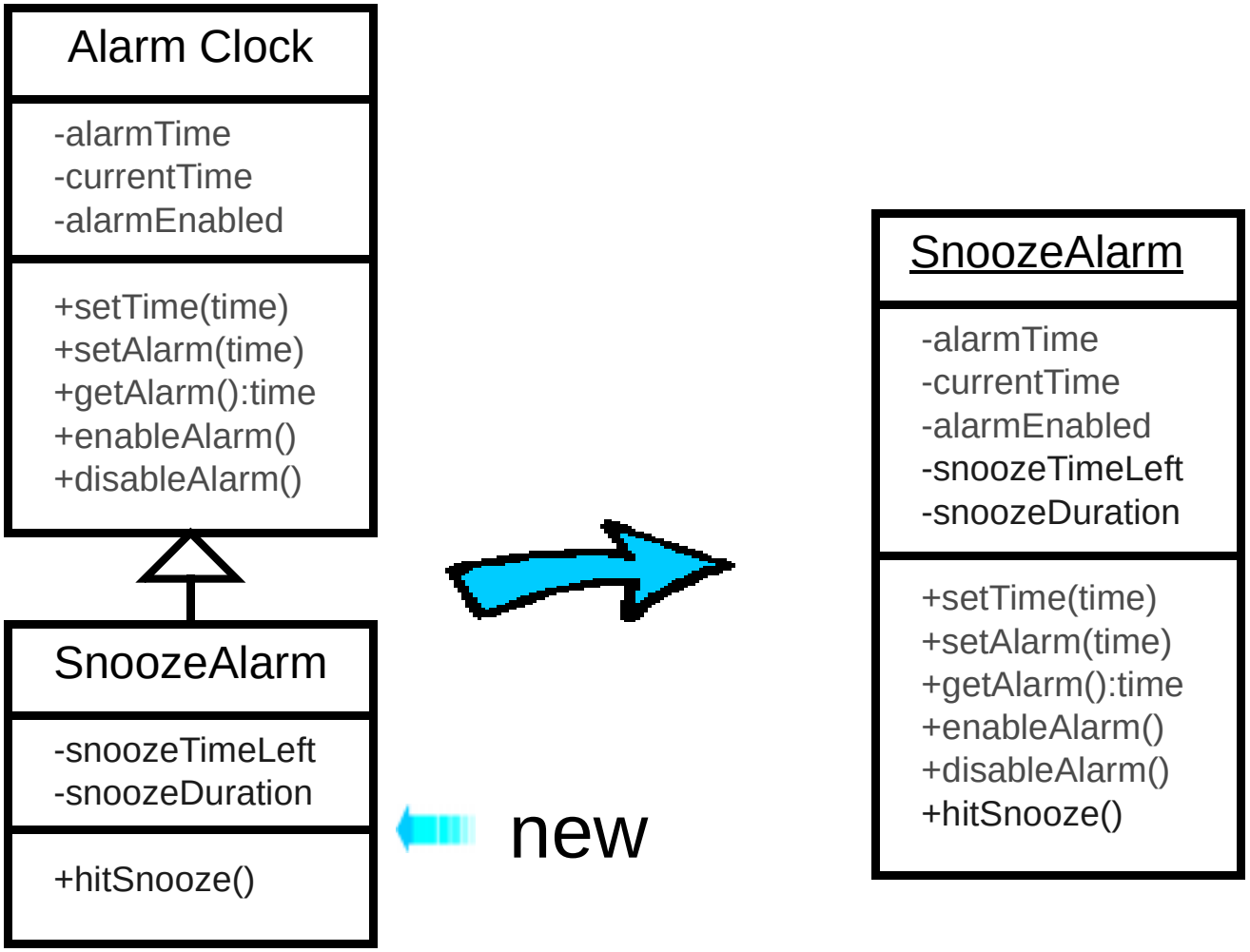
Subclass
Derived class
Child class

Inheritance

- A key test is the IS-A or IS-A-KIND-OF rule. Ask "Does it make sense to say that A is a B (or A is a kind of B)? If so, you may have discovered an inheritance relationship. If not, the relationship is probably not inheritance.

Examples: A ManufacturedWidget is a Widget. SmallCustomer is a kind of Customer.

Inheritance



Implications For Testing

- Changes to one class definition can be (almost) automatically propagated to several others.
- Some definitions and declarations are reused in succeeding layers and interact with new definitions and declarations.
- There is potential for savings of effort if testing is sufficiently organized.
- But, the basic “is-a” relationship must *not* be violated.

Exercise 2.3

Exercise

- Determine the relationships (if any) between the following classes:

City and City Hall _____

Planet and Class “M” Planet _____

Continent and Country _____

Country and City _____

Democracy and Government _____

Monarch and Country _____

U.S. Senate and Senator _____

State and Province _____

Definitions

- Pre-condition – The pre-condition for a method is a statement, that may be either true or false, of the assumptions made in designing the method. For example, the Add method for a list could have a pre-condition that no Add will be attempted if the list is full.

- Post-condition – The post-condition for a method is a statement of the results produced by a method, provided the pre-condition was true initially. For example, the Add method for a list would have a post-condition that the item has been added to the list, assuming the list was not full initially.

Definitions

- Invariant – A statement of those things we assume to be true about the class at all times. For example, a queue has an invariant that the items placed into the queue will be returned in the same order and will be unchanged.

Design By Contract

```
Class Queue{
    public:
        Pre: None
        Queue();
        Post: A queue instance exists, is empty, and has capacity

        Pre: !queue.isFull()
        enqueue(Itemtype x);
        Post: (queue'.length() == queue.length() + 1)
            && (queue'.contains(x))

        Pre: !queue.isEmpty()
        dequeue() returns itemtype;
        Post: (queue'.length() == queue.length() - 1)
            && return x such that queue.contains(x))

        Pre: None
        isEmpty() returns boolean;
        Post: returns (queue.length() == 0)
    };
invariant: 0 <= length <= capacity
```

Defensive Design

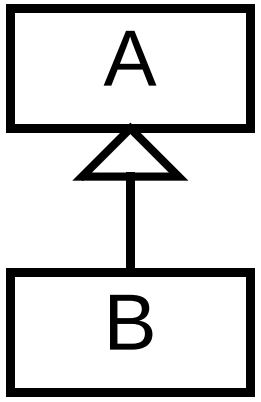
```
Class Queue{
    public:
        Pre: None
        Queue();
        Post: A queue instance exists, is empty, and has capacity

        Pre: None
        enqueue(Itemtype x) throws OVERFLOW;
        Post: if queue.isFull() then OVERFLOW thrown
            else (queue'.length() == queue.length() + 1)
            && (queue'.contains(x))

        Pre: None
        dequeue() returns itemtype throws UNDERFLOW;
        Post: if queue.isEmpty() then UNDERFLOW thrown
            else (queue'.length() == queue.length() - 1)
            && return x such that queue.contains(x))

        Pre: None
        isEmpty() returns boolean;
        Post: returns (queue.length() == 0)
};
invariant: 0 <= length <= capacity
```

Strict Inheritance

Define

- Strict inheritance means using inheritance properly, i.e., not violating the is-a relationship.

All objects of class B must also be objects of class A.



The protocol of A must be contained within the protocol of B.

- Whenever I say “inheritance” I mean “strict inheritance”.

Strict Inheritance

- Strict inheritance means:
  - Pre-conditions on a particular method in a derived class must be the same or weaker than those of the same method in the parent class.
  - Post-conditions on a particular method in a derived class must be the same or stronger than those on the same method in the parent class.
  - The invariant for a class must be the same or stronger than the invariant for the parent class.

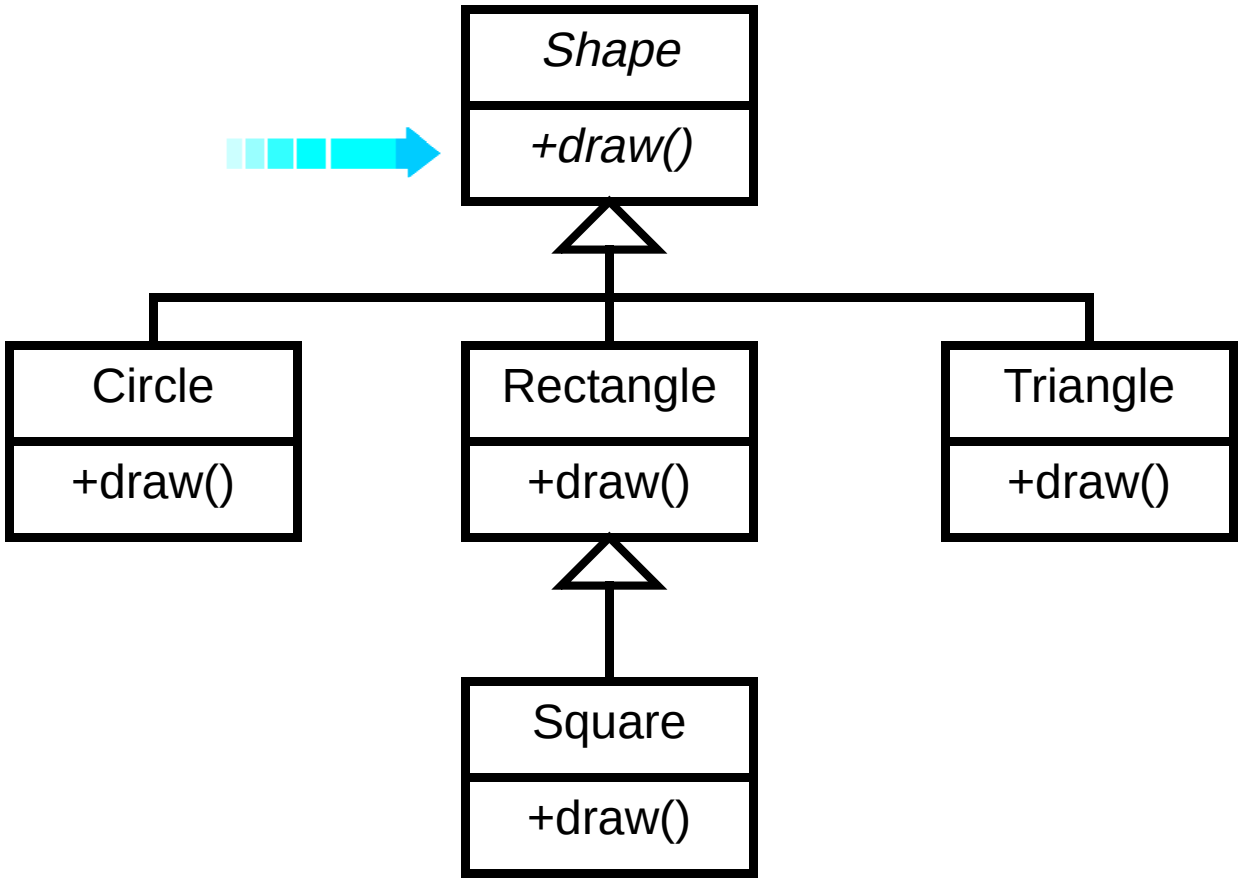
Polymorphism

Define

- Polymorphism is the ability to send a message to an object without having to know the specific class of the object.
- Each object knows its class and thus knows how it should respond to messages sent to it.
- Polymorphism can be used whenever several classes within an inheritance structure share a common protocol.

Polymorphism

Document



Polymorphism

Develop

Without Polymorphism

```
// Given an object x

if(x.getClass()==”A” {
   doAStuff():
}
if(x.getClass()==”B” {
   doBStuff():
}
if(x.getClass()==”C” {
   doCStuff():
}
…
```

With Polymorphism

```
// Given an object x

x.doStuff();
```

Which would you rather write? Which would you rather maintain?

Implications For Testing

- Each class is a package of methods and attributes that must be tested as a unit.
- There are more inter-object connections to consider due to polymorphism. Traditional static analysis does not apply here.
- There are more methods to manage. Each method may send several messages to other objects. This increases system complexity.
- Each method is smaller than typical procedural functions/subroutines. This decreases system complexity.

Summary

-
-
-
-
-

Exercise 2.4

Exercise

Background

Welcome to the Software Architects rent-a-car agency. We've gathered together the following information from interviews, documents, and observations about our company. Some of the descriptions are in use-case (scenario) format. Additional data is presented in organized lists for your analysis.

Required Data

Reservation

The information needed for a reservation includes:

- Planned pickup location, date, and time
- Planned return location, date, and time
- Vehicle type desired (compact, luxury, van,...)

Exercise 2.4

- Renter's name or Renter's Express Membership number
- Renter's arrival airline
- Reservation number

Renter

The information kept for a renter includes:

- Renter's name, address, phone number, drivers license number
- Renter's Express Membership number (if available)

Express Member

The information kept for an Express Member includes:

Exercise 2.4

- Member's name, address, phone number, drivers license number
- Membership number
- Preferences about size and model of car
- Credit card number usually used

Contract

Information maintained as part of a contract includes:

- Actual pickup location, date, and time
- Planned return location, date, and time
- Actual return location, date, and time
- Vehicle rented
- Payment type (cash, credit card, corporate account)

Exercise 2.4

- Renter's name, address, phone number, drivers license number OR Renter's Express Membership number
- Rental rate
- Contract number

Vehicle

The information kept for each Vehicle includes:

- Make, model, year
- Vehicle Identification Number (VIN)
- Serial number
- Mileage
- Type (compact, standard, luxury, van,...)

Exercise 2.4

Use Case #1

A person interacts with a reservation agent to create a reservation.

- Person calls the reservation agent to request a reservation
- Reservation agent requests the following information:
  - Planned pickup location, date, and time
  - Planned return location, date, and time
  - Desired vehicle class
  - Arrival airline
  - Renters express number OR name if no express number
- System checks for vehicle class availability, locates member's express record, verifies locations
- System determines the rental rate

Exercise 2.4

- Reservation agent reviews information with caller and gets approval
- System saves (books) the reservation
- Reservation agent gives the caller the reservation number

Use Case #2

A person with a reservation comes to the rental office to rent a vehicle

- A person (renter) comes to the rental location
- The system locates the reservation
- A contract is created
- Information from the reservation is moved into the contract
- The estimated total cost is computed

Exercise 2.4

- The payment method is determined
  - If cash then take a deposit
  - If credit card then take an imprint and place a hold on the renter's credit
  - If a corporate account then take the account number and verify
- Assign a vehicle from the inventory
- Print the initial copy of the contract and give it to the renter

Use Case #3

A renter returns a vehicle

- Locate the contract

Exercise 2.4

- Record the actual return location, date, and time
- Compute the actual charges
- Return the vehicle to inventory
- Print the final copy of the contract

Assignment

Given the description above, create a class model based on the important classes and their interrelationships. Prepare a transparency for presentation to the class.