

Das and
Sourangshu
Bhattacharya

Inheritance

rielper Classe

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Refactoring

Hierarchy Integration

Module Summar

# Module 24: Programming in C++

Part 4: Phone Hierarchy

#### Intructors: Abir Das and Sourangshu Bhattacharya

Department of Computer Science and Engineering Indian Institute of Technology, Kharagpur

{abir, sourangshu}@cse.iitkgp.ac.in

Slides taken from NPTEL course on Programming in Modern C++

by Prof. Partha Pratim Das



## Module Objectives

Das and
Sourangshu

ISA Hierarchy I

Helper Class

Hierarchy of Phones by

Interfaces &
State Variables of

Landline Phone

Refactoring

Hierarchy Integration

Module Summar

• Model a hierarchy of phones using inheritance



### Module Outline

Intructors: Abi Das and Sourangshu Bhattacharya

ISA Hierarchy b Inheritance

Helper Class

Hierarchy o Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone Mobile Phone Smart Phone

Refactorin

Hierarchy Integration Extended Hierarc of Phones

Module Summar

- ISA Hierarchy by Inheritance
- 2 Helper Classes
- 3 Hierarchy of Phones by Interfaces
- 4 Interfaces & State Variables of Phones
  - Landline Phone
  - Mobile Phone
  - Smart Phone
- 6 Refactoring
- 6 Hierarchy Integration
  - Extended Hierarchy of Phones
- Module Summary



## Approach to Modeling Hierarchy

Intructors: Abir Das and Sourangshu Bhattacharya

ISA Hierarchy by Inheritance

Helper Class

Hierarchy o Phones by Interfaces

Interfaces & State Variables o Phones

Landline Phone
Mobile Phone
Smart Phone

Refactorin

Hierarchy Integration Extended Hierarch of Phones

Module Summa

- Identify the **Concepts and their ISA relationships** to define the hierarchy: model with public inheritance
- Identify and model **Helper classes** lower level UDTs to define components
- Identify the Interface of each concept: signatures of public member functions
- Identify the **State Variables** of each concept: types of of private / protected data members (also member functions used for ease of implementation)
- **Refactor** common data members and member functions between specialized and generalized classes to link the classes by inheritance
- Integrate the hierarchy with abstract (pure) interface
- Explore extendability
- We illustrate with the phone hierarchy



## Helper Classes

class Photo

class RingTone

class AddressBook

class Contact

Intructors: Abir Das and Sourangshu Bhattacharya

ISA Hierarchy b Inheritance

#### Helper Classes

Hierarchy o Phones by Interfaces

Interfaces & State Variables of Phones

Mobile Phone

Refactorin

Hierarchy Integration Extended Hier

Module Summai

### **Class** Description

class PhoneNumber 12-digit phone number class Name Subscriber Name (as s:

Subscriber Name (as string)

Image & Subscriber Name as alt text

Audio & ring tone name

PhoneNumber, Name, and Photo (optional) of a contact

List of contacts



### Hierarchy of Phones

Intructors: Abi Das and Sourangshu Bhattacharya

ISA Hierarchy b Inheritance

Helper Class

Hierarchy of Phones by Interfaces

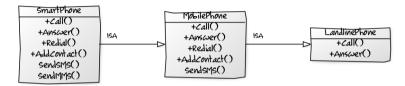
Interfaces & State Variables ( Phones

Landline Phone
Mobile Phone
Smart Phone

Refactorin

Hierarchy Integration Extended Hierarc

Module Summa



- MobilePhone ISA LandlinePhone
  - o LandlinePhone is generalization
  - o MobilePhone is specialization
  - MobilePhone inherits the properties of LandlinePhone
- SmartPhone ISA MobilePhone
  - o MobilePhone is generalization
  - SmartPhone is specialization
  - SmartPhone inherits the properties of MobilePhone
- ISA is transitive



### Interfaces of Phones

Intructors: Abii Das and Sourangshu Bhattacharya

ISA Hierarchy by Inheritance

Helper Classe

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone
Mobile Phone
Smart Phone

Refactorin

Hierarchy Integration Extended Hierard of Phones

Module Summar

#### • Landline Phone

- Call: By dial / keyboard
- Answer
- Caller ID (with special attached device)

#### Mobile Phone

- Call: By keyboard shows number
  - ▷ By Number
  - ▷ By Name
- Answer
- o Caller ID
- Redial

CS20202: Software Engineering

- Set Ring Tone
- Add Contact
  - ▶ Number
  - Name

#### Smart Phone

- Call: By touchscreen shows number & photo
  - ▷ By Number
  - ▷ By Name
- Answer
- Caller ID
- o Redial
- Set Ring ToneAdd Contact
- Number
  - Numbe
  - ▶ Name
  - ▷ Photo

- There exists a substantial overlap between the functionalities of the phones
- A mobile phone is more capable than a land line phone and can perform (almost) all its functions
- A smart phone is more capable than a mobile phone and can perform (almost) all its functions
- These phones belong to a Specialization / Generalization Hierarchy



#### Interface & State Variable: Landline Phone

Intructors: Ab
Das and
Sourangshu
Bhattacharya

Inheritance

Helper Class

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Refactoring

Hierarchy Integration

Module Summar

```
class LandlinePhone {
    PhoneNumber number_;
    Name subscriber_;
    RingTone rTone_;

public:
    LandlinePhone(const char *num, const char *subs);
    void Call(const PhoneNumber *p);
    void Answer();
    friend ostream& operator<<(ostream& os, const LandlinePhone& p);
};</pre>
```

**Landline Phone** 

Answer

Call: By dial / keyboard



#### Interface & State Variable: Mobile Phone

Intructors: Abi Das and Sourangshu Bhattacharya

ISA Hierarchy by Inheritance

Helper Classe

Hierarchy of Phones by Interfaces

Interfaces & State Variables o Phones

Mobile Phone
Smart Phone

Refactorin

Hierarchy Integration Extended Hierarch

Module Summar

```
    Mobile Phone
```

- Call: By keyboard shows number
  - ▷ By Number
  - ⊳ By Name
- Answer
- o Redial
- Set Ring Tone
- Add Contact
  - Number
  - > Name

```
class MobilePhone {
    PhoneNumber number :
    Name subscriber :
    RingTone rTone_;
    AddressBook aBook :
    PhoneNumber *lastDial_;
    void SetLastDialed(const PhoneNumber& p);
    void ShowNumber();
public:
    MobilePhone(const char *num, const char *subs);
    void Call(PhoneNumber *p):
    void Call(const Name& n):
    void Answer():
    void ReDial():
    void SetRingTone(RingTone::RINGTONE r):
    void AddContact(const char *num = 0.
        const char *subs = 0):
    friend ostream& operator << (ostream& os. const MobilePhone& p):
};
```



### Interface & State Variable: Smartphone

Intructors: Abi Das and Sourangshu Bhattacharya

ISA Hierarchy by Inheritance

Helper Classe

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone
Mobile Phone
Smart Phone

Refactorin

Hierarchy Integration Extended Hierard

Module Summar

```
Smart Phone
```

- Call: By touchscreen shows number & photo
  - ▷ By Number
  - By Name
- Answer
- Redial
- Set Ring Tone
  - Add Contact
    - ▷ Number
    - ▷ Name
    - ▷ Photo

```
class SmartPhone {
    PhoneNumber number :
    Name subscriber :
    RingTone rTone_;
    AddressBook aBook :
    PhoneNumber *lastDial :
    void SetLastDialed(const PhoneNumber& p):
    void ShowNumber():
    unsigned int size :
    void DisplayPhoto():
public:
    SmartPhone(const char *num, const char *subs):
    void Call(PhoneNumber *p):
    void Call(const Name& n):
    void Answer():
    void ReDial():
    void SetRingTone(RingTone::RINGTONE r);
    void AddContact(const char *num = 0.
        const char *subs = 0):
    friend ostream& operator << (ostream& os. const MobilePhone& p):
              Intructors: Abir Das and Sourangshu Bhattacharva
```



## Refactoring

Das and

SA Hierarchy

Helper Class

Hierarchy of Phones by

Interfaces &
State Variables

Landline Phone

Refactoring

Hierarchy Integration

Extended Hier of Phones

Module Summa

### Refactoring



#### MobilePhone ISA LandlinePhone

Intructors: Ab
Das and
Sourangshu
Bhattacharya

ISA Hierarchy by

Helper Classe

Hierarchy o Phones by Interfaces

Interfaces & State Variables o Phones

Landline Phone
Mobile Phone
Smart Phone

Refactoring

Hierarchy Integration Extended Hierarch of Phones

Module Summa

```
class MobilePhone : public LandlinePhone { protected:
class LandlinePhone { protected:
    PhoneNumber number :
                                                       //PhoneNumber number :
   Name subscriber :
                                                       //Name subscriber :
   RingTone rTone :
                                                       //RingTone rTone :
                                                       AddressBook aBook :
                                                       PhoneNumber *lastDial :
                                                       void SetLastDialed(const PhoneNumber& p):
                                                       void ShowNumber():
public:
                                                  public:
    LandlinePhone(const char *num.
                                                       MobilePhone(const char *num.
        const char *subs) :
                                                           const char *subs) :
        number (num), subscriber (subs),
                                                           LandlinePhone(num. subs), // Base ctor
        rTone_() { }
                                                           lastDial_(0) { }
    void Call(const PhoneNumber *p);
                                                       void Call(const PhoneNumber *p): // Override
                                                       void Call(const Name& n):
                                                                                        // Overload
   void Answer():
                                                       //void Answer():
                                                                                         // Inherited
                                                       void ReDial():
                                                       void SetRingTone(RingTone::RINGTONE r);
                                                       void AddContact(const char *num = 0.
                                                           const char *subs = 0):
                                                       friend ostream& operator << (ostream& os.
   friend ostream& operator << (ostream& os.
        const LandlinePhone& p);
                                                           const MobilePhone& p);
};
                                                   };
```



#### MobilePhone ISA LandlinePhone

Intructors: Abi Das and Sourangshu Bhattacharva

Inheritance

Helper Classe

Hierarchy o Phones by Interfaces

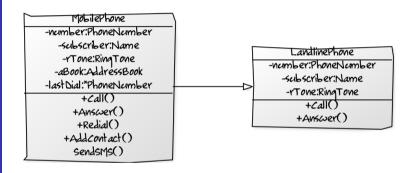
Interfaces & State Variables o Phones

Landline Phone
Mobile Phone

#### Refactoring

Hierarchy Integration Extended Hierar

Module Summai





#### SmartPhone ISA MobilePhone

Refactoring

```
class MobilePhone : public LandlinePhone { protected: class SmartPhone : public MobilePhone { protected:
    //PhoneNumber number :
                                                           //PhoneNumber number :
    //Name subscriber :
                                                           //Name subscriber :
    //RingTone rTone :
                                                           //RingTone rTone :
    AddressBook aBook :
                                                           //AddressBook aBook :
    PhoneNumber *lastDial :
                                                           //PhoneNumber *lastDial :
                                                           //void SetLastDialed(const PhoneNumber& p);
    void SetLastDialed(const PhoneNumber& p);
    void ShowNumber():
                                                           //void ShowNumber():
                                                           unsigned int size_;
                                                           void DisplayPhoto()
public:
                                                       public:
   MobilePhone(const char *num.
                                                           SmartPhone(const char *num.
        const char *subs) :
                                                                const char *subs) :
        LandlinePhone(num, subs), // Base ctor
                                                                MobilePhone(num, subs), // Base ctor
        lastDial (0) { }
                                                                lastDial (0) { }
    void Call(const PhoneNumber *p): // Override
                                                           void Call(const PhoneNumber *p): // Override
    void Call(const Name& n):
                                                           void Call(const Name& n):
                                     // Overload
                                                                                             // Override
    //void Answer():
                                     // Inherited
                                                           //void Answer():
    void ReDial():
                                                           void ReDial():
                                                                                             // Override
    void SetRingTone(RingTone::RINGTONE r);
                                                           //void SetRingTone(RingTone::RINGTONE r);
    void AddContact(const char *num = 0.
                                                           //void AddContact(const char *num = 0.
        const char *subs = 0);
                                                                //const. char *subs = 0):
   friend ostream& operator<< (ostream& os.
                                                           friend ostream& operator<< (ostream& os.
        const MobilePhone& p);
                                                                const SmartPhone& p);
};
                                                        };
```



## Hierarchy Integration

Das and Sourangshu

ISA Hierarchy Inheritance

Helper Class

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Refactorin

Hierarchy Integration

Extended Hiera of Phones

lodule Summ

### **Hierarchy Integration**



### Hierarchy Integration

class Phone { public:

// Abstract Base Class - A Pure Interface

virtual void Call(const PhoneNumber \*p) = 0;

Intructors: Abir Das and Sourangshu Bhattacharya

ISA Hierarchy by Inheritance

Helper Classes

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Mobile Phone
Smart Phone

rteractoriii

Hierarchy Integration Extended Hierarch

Module Summary

```
virtual void Answer() = 0:
    virtual void ReDial() = 0:
};
class LandlinePhone: public Phone {
protected:
    PhoneNumber number :
    Name subscriber :
    RingTone rTone_;
public:
    LandlinePhone(const char *num.
        const char *subs) :
        number (num), subscriber (subs),
        rTone_() { }
    // Implementations for interfaces
    void Call(const PhoneNumber *p);
    void Answer():
    // Dummy implementation not for use
    void ReDial()
    { cout << "Not implemented" << endl: }
    friend ostream& operator << (ostream& os,
        const LandlinePhone& p);
  CS20202: Software Engineering
```

```
class MobilePhone : public LandlinePhone { protected:
    AddressBook aBook :
    PhoneNumber *lastDial :
    void SetLastDialed(const PhoneNumber& p):
    void ShowNumber():
public:
    MobilePhone(const char *num, const char *subs) :
       LandlinePhone(num, subs), lastDial_(0) { }
    void Call(const PhoneNumber *p); // Override
    void Call(const Name& n):
                                     // Overload
    void ReDial();
                                     // Override
    friend ostream& operator << (ostream& os,
       const MobilePhone& p):
class SmartPhone : public MobilePhone {
protected: unsigned int size_;
    void DisplayPhoto():
public:
    SmartPhone(const char *num, const char *subs) :
       MobilePhone(num, subs), lastDial (0) { }
    void Call(const PhoneNumber *p); // Override
    void Call(const Name& n):
   void ReDial():
                                     // Override
    friend ostream& operator << (ostream& os,
       const SmartPhone& p);
```

Intructors: Abir Das and Sourangshu Bhattacharva



### Extended Hierarchy of Phones

Intructors: Abi Das and Sourangshu Bhattacharva

ISA Hierarchy Inheritance

Helper Class

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

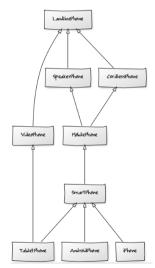
Landline Phon Mobile Phone

Refactoring

Hierarchy Integration

Extended Hierarchy of Phones

Module Summai





## Module Summary

Intructors: Ab
Das and
Sourangshu
Bhattacharya

Inheritance

Helper Class

Hierarchy of Phones by Interfaces

Interfaces & State Variables Phones

Landline Phore

Refactoring

Hierarchy Integration Extended Hie

Module Summary

• Using the Phone Hierarchy as an example analyzed the design process with inheritance