Java in Bioinformatics

Gang Chen
chengang@bgitechsolutions.com

October 25, 2014

华大科技 BGI·tech
香港中文大學 The Chinese University of Hong Kong

Outline

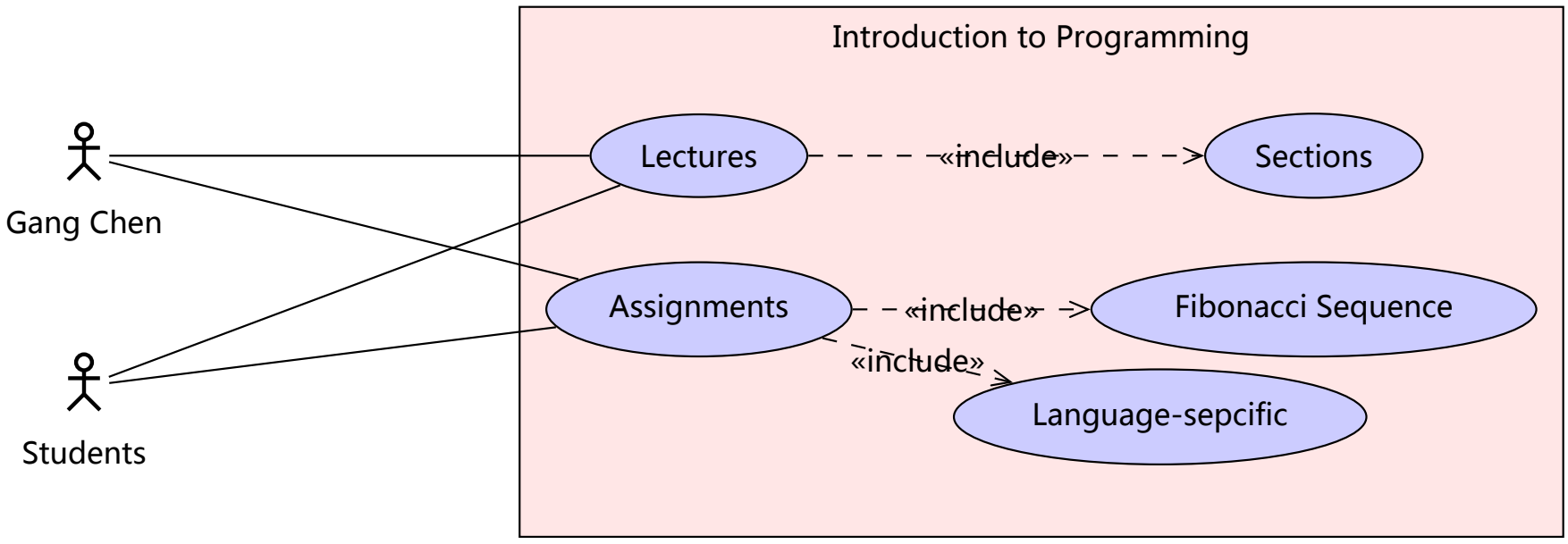
Next

Next

Object-oriented Programming (OOP)

- OO is a language-indedependent concept
- in principle not limited to programming (OO design)
  $\Longrightarrow$ databases, business plans
- improves reusability and exchangability of code
- separation of partial problems
- ``real world'' modelling
- representation in Universal Markup Language (UML)

Object-oriented Programming (OOP)



Basic ideas

- Everything is an object
  Gang, students, lecture
  project, world, spy, information
- Objects interact by sending/receiving messages
  Gang $\rightarrow$ students: object orientation is a concept
  world $\rightarrow$ map: what is the object at position X?
- An object consists of objects
  a course consists of lectures
  a world consists of land or water fields

Basic ideas

- Every object has a type
  Gang is a teacher
  the map is a rectangle of land / water fields
- All objects of the same type understand the same messages
  all students hear the lecture
  all spies can retrieve information

Next

Classes, Interfaces and Methodse

World
spies : ListOfSpies
setSpyCount(count: int) getSpyCount() : int getMap() : Map

class

attributes

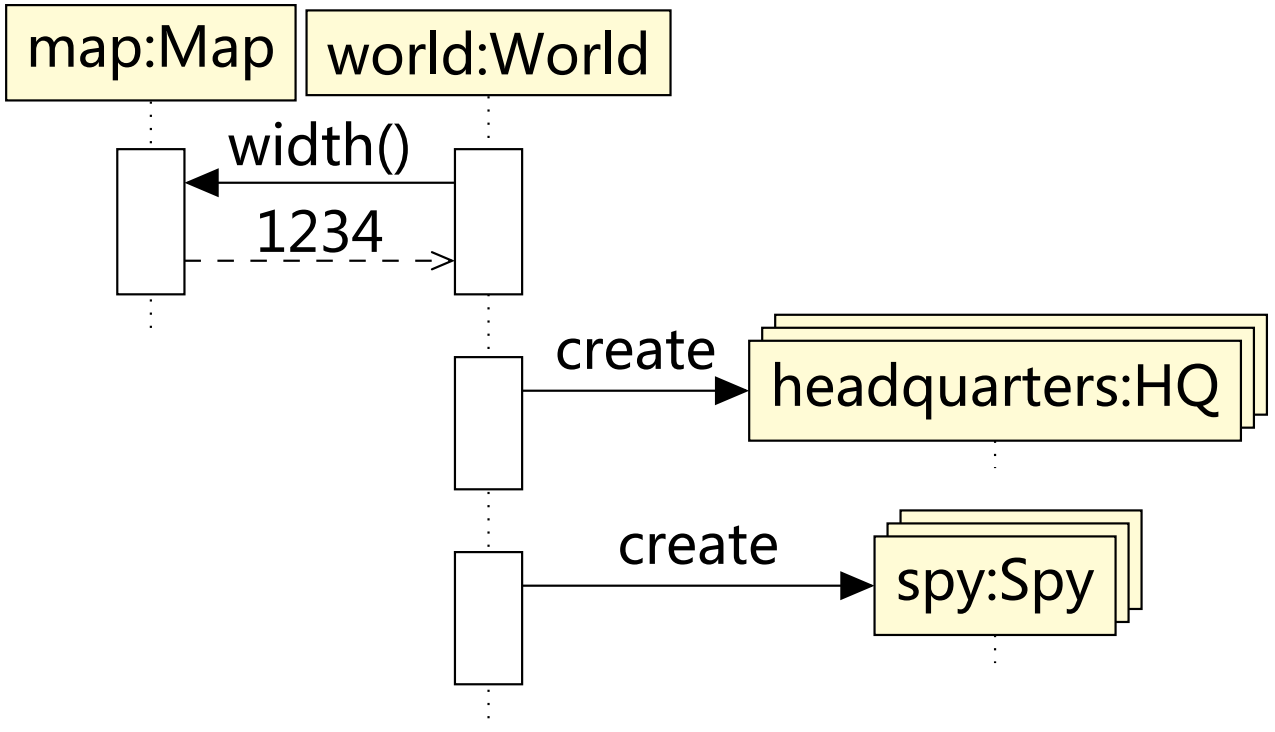
methods

Map
tiles: VectorOfTiles
getWidth() : int getHeight() : int at(x: int, y:int) : Tile

Classes, Interfaces and Methods

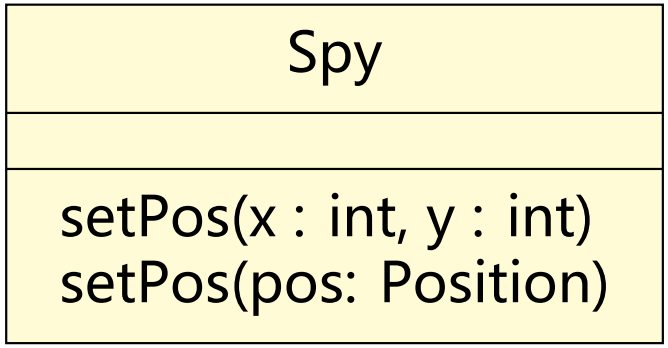
- classes describe the type of objects (define their interface)
- the interface consists of methods and attributes / properties
- methods
  - are functions that operate on objects of this class
  - can take extra arguments of arbitrary types
  - can return values of arbitrary types
- attributes are objects of arbitrary other types

Objects/Instances



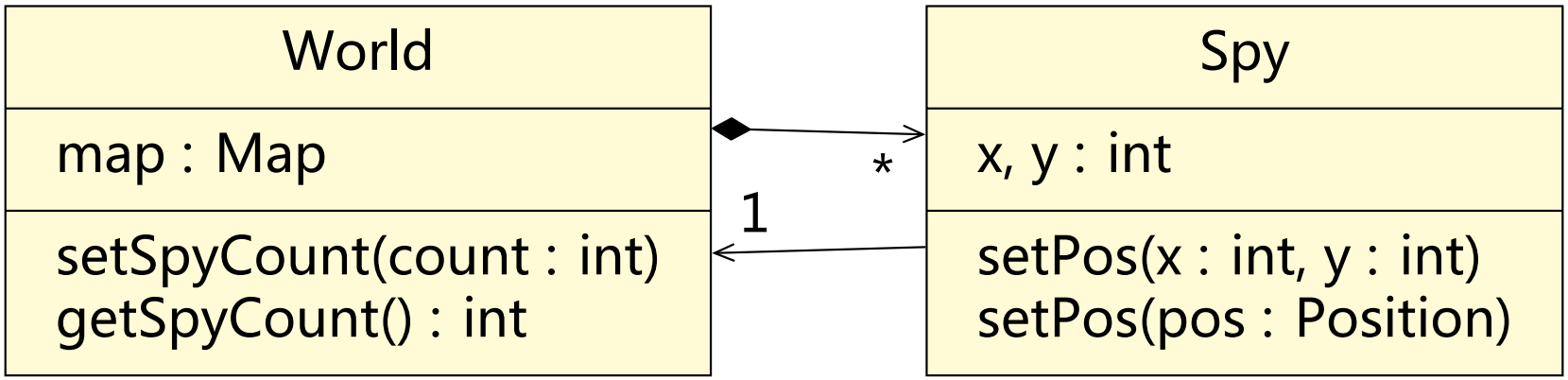
- every object has an immutable class assigned when it is created
- objects communicate via their class interfaces
- classes can communicate via static member functions

Overloading and signature



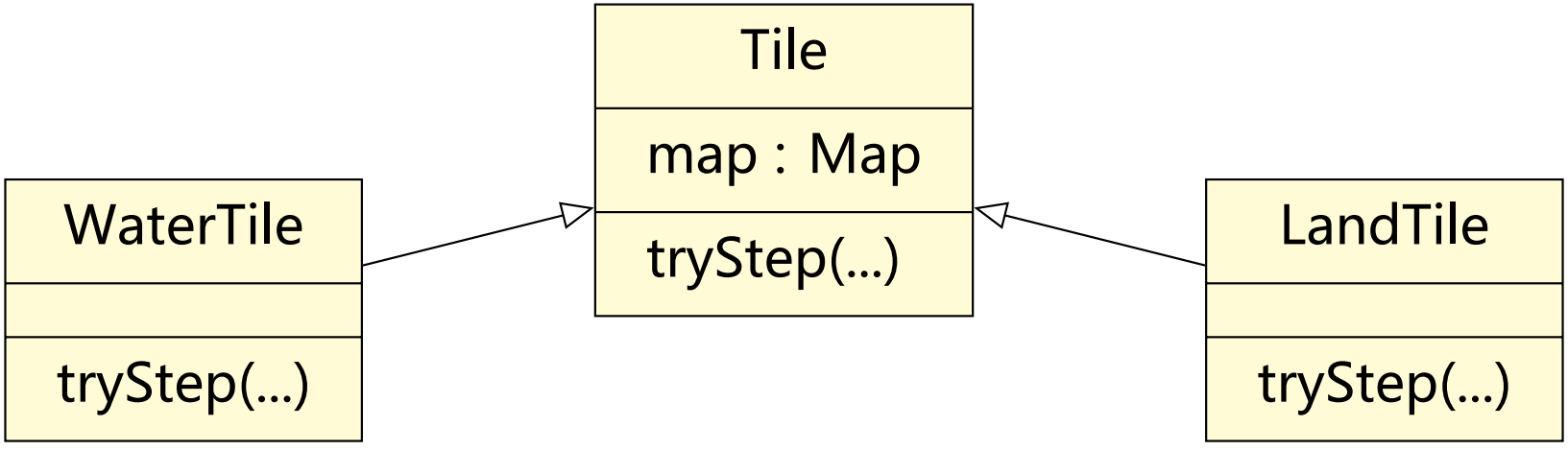
- a method is described by name and signature
- signature is formed by the types of all taken arguments
  - setPos(x : int, y : int)
  - setPos(pos : Position)
- methods with identical names but different arguments can exist in one class -- overloading
- return type is not part of the signature -- cannot always resolve overload at compile time

Composing classes



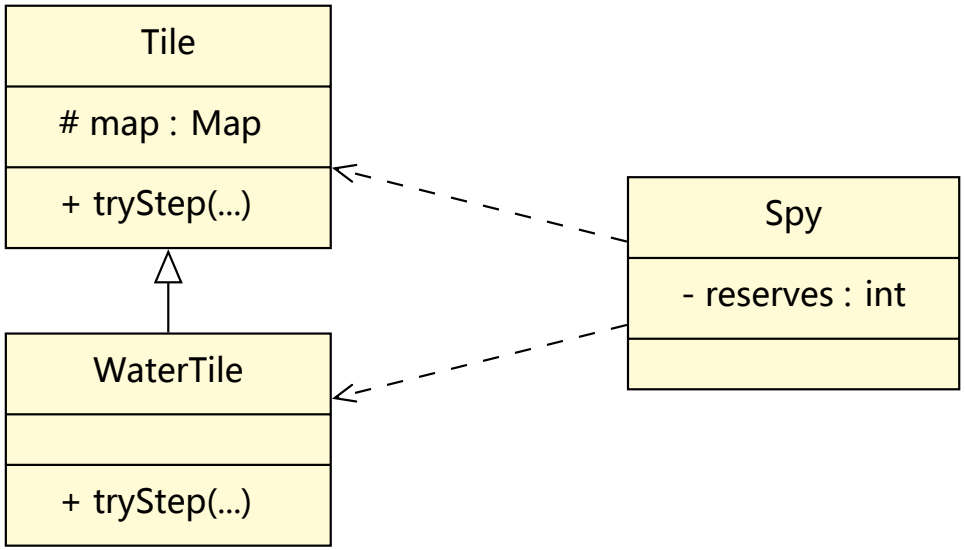
- objects are made of objects (attributes) --- classes declare the types of these objects
- simple attributes appear below class name
- complex classes shown as composition
- ``has-a'' or ``has-many'' relations:
  - a world hosts many spies,
  - a spy belongs to one world

Inheritance and class hierarchy



- subclasses of classes -- class hierarchy
- subclasses inherit methods and attributes of all superclasses
- no need to duplicate code
- .. but methods might behave differently (polymorphism)
- abstract classes implement only parts of the interface

Implementation hiding



- public (`+') elements are visible to all
- protected (`#') elements are only visible to derived classes
- private (`-') attributes or methods are not visible to other objects
- map is protected $\Longrightarrow$ visible to WaterTile, not to Spy
- Spy can access tryStep of all tiles

Next

Hello Java

```
public class HelloJava{

    public static void main(String[] args) {
        System.out.println("Hello Java!");
    }

}
```

Next

Books

Books

- Thinking in Java
- Core Java
- Martin Fowler, ``UML Distilled'', 3rd edition, Addison-Wesley

Next

GATK

The Genome Analysis Toolkit or GATK is a software package developed at the Broad Institute to analyze high-throughput sequencing data. The toolkit offers a wide variety of tools, with a primary focus on variant discovery and genotyping as well as strong emphasis on data quality assurance. Its robust architecture, powerful processing engine and high-performance computing features make it capable of taking on projects of any size.

GATK

https://www.broadinstitute.org/gatk/

Next

Cytoscape

Cytoscape is an open source software platform for visualizing complex networks and integrating these with any type of attribute data. A lot of Apps are available for various kinds of problem domains, including bioinformatics, social network analysis, and semantic web.

Cytoscape

http://www.cytoscape.org/

Cytoscape

- Documentation: http://opentutorials.cgl.ucsf.edu/index.php/Portal:Cytoscape3
- Source codes: https://github.com/cytoscape/cytoscape-impl
- User Manual in Chinese: https://code.google.com/p/cytoscape-cn/

Apps

- Developer: http://wiki.cytoscape.org/Cytoscape_3/AppDeveloper
- Example: ClusterViz