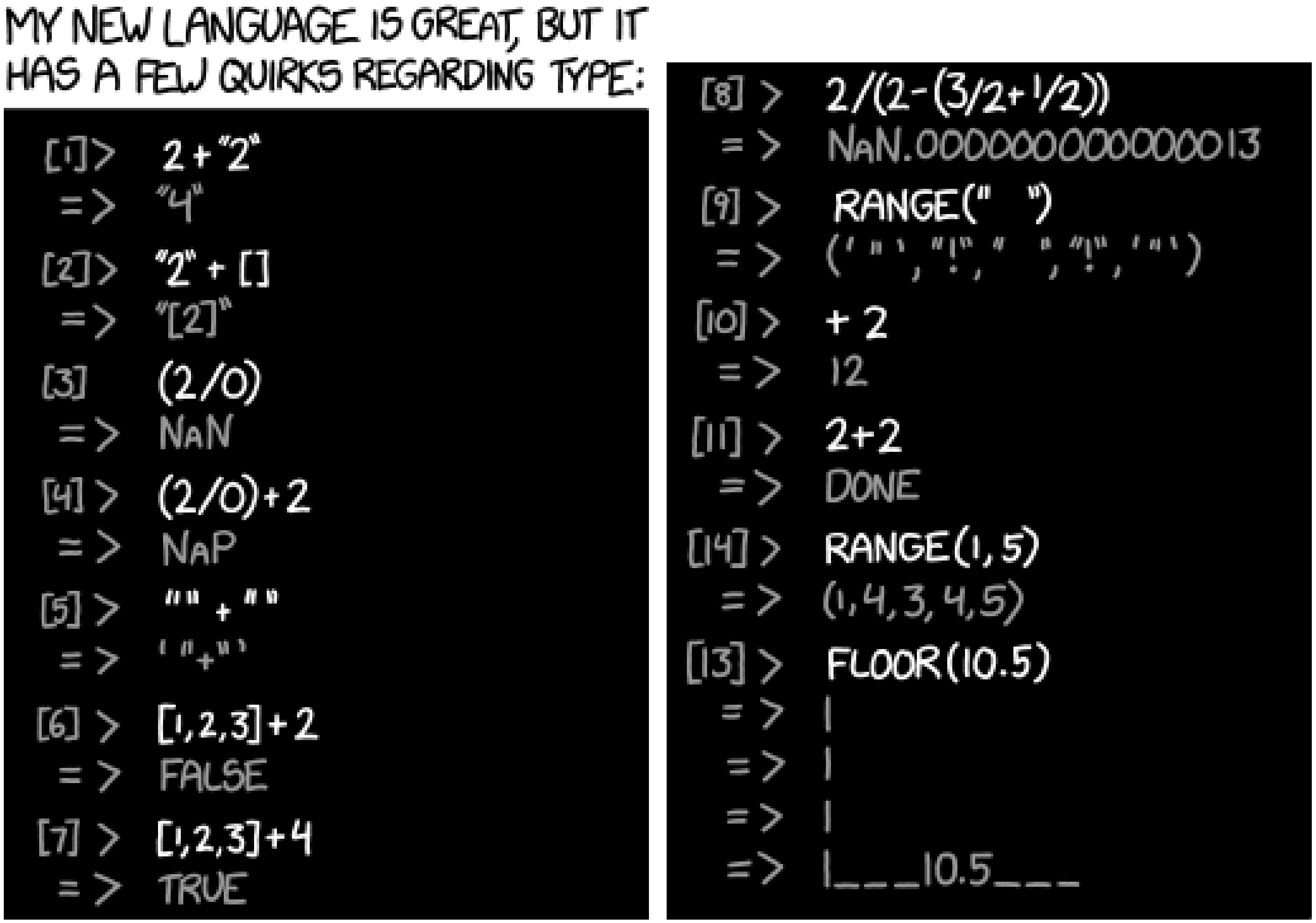


MY NEW LANGUAGE IS GREAT, BUT IT HAS A FEW QUIRKS REGARDING TYPE:

```
[1]>  2+"2"
  =>  "4"
[2]>  "2"+[]
  =>  "[2]"
[3]   (2/0)
  =>  NaN
[4]>  (2/0)+2
  =>  NaP
[5]>  ""+""
  =>  '"+"'
[6]>  [1,2,3]+2
  =>  FALSE
[7]>  [1,2,3]+4
  =>  TRUE
```

```
[8]>  2/(2-(3/2+1/2))
  =>  NaN.0000000000000013
[9]>  RANGE(" ")
  =>  ('"',"!"," ","!",'"')
[10]> +2
  =>  12
[11]> 2+2
  =>  DONE
[14]> RANGE(1,5)
  =>  (1,4,3,4,5)
[13]> FLOOR(10.5)
  =>  |
  =>  |
  =>  |
  =>  |___10.5___
```

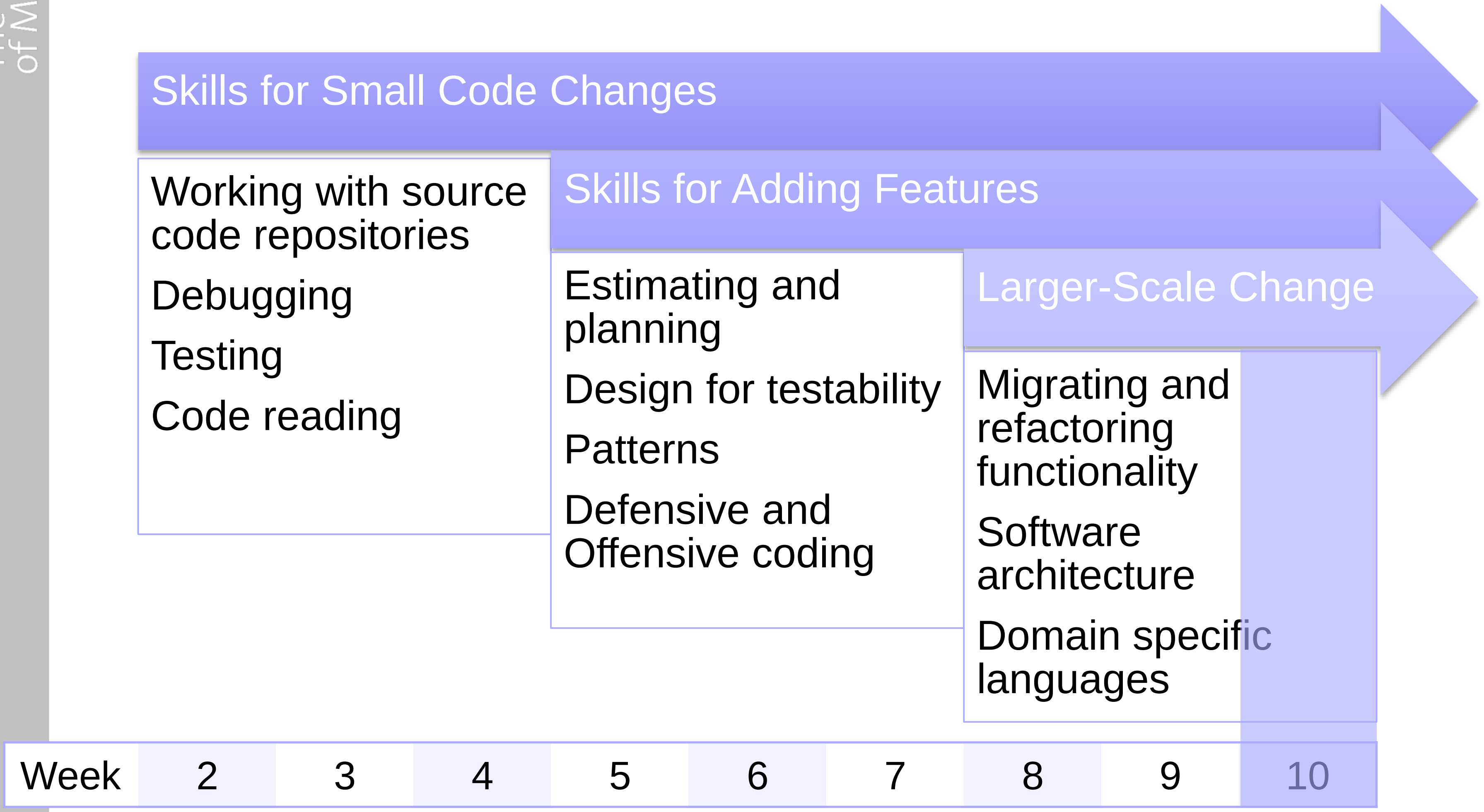
Domain Specific Languages and the Practice Exam

COMP23420: Software Engineering

Week 10

Robert Haines and Caroline Jay

Course Unit Roadmap (Weeks 2-10)



Skills for Small Code Changes

- Working with source code repositories
- Debugging
- Testing
- Code reading

Skills for Adding Features

- Estimating and planning
- Design for testability
- Patterns
- Defensive and Offensive coding

Larger-Scale Change

- Migrating and refactoring functionality
- Software architecture
- Domain specific languages

Week	2	3	4	5	6	7	8	9	10

Link to the Coursework/Exam

- We will introduce Domain Specific Languages and discuss the practice exam paper.
- Coursework: Link the final exercise with real-world DSL examples.
- Exam: There will be questions on DSLs in the exam.

What is a domain?

- Area of interest
- Scientific specialty

- Bounded
- Overlapping

- Examples
  - Computer Science
    - Software Engineering
    - Human Computer Interaction

Domain Specific Languages

- Domain Specific Languages are very common
  - They underpin many key technologies
- Also known as
  - Little languages
  - Minilanguages
- Mostly textual
  - Ideally more natural looking than general code

DSL examples

- Web
  - HTML, CSS, JavaScript, XSLT
- Development tools
  - make, rake, ant, lex, yacc, Emacs Lisp
- Databases
  - SQL, HQL, Object/Relational Mapping in general
- Typesetting utilities
  - TeX, LaTeX, troff, groff, PostScript
- Unix tools
  - sed, awk, bc, m4
- Other
  - Regular expressions, Office macros, MATLAB

DSLs vs general-purpose languages

- Design goals contrast those of general languages

- Less complex/comprehensive
  - Smaller syntax (fewer opportunities for bugs?)
  - Focussed on a particular domain

- More expressive
  - Optimized for tasks within a domain

- Both of the above mean a DSL is unlikely to be of general use outside their domain

Why use a DSL?

- Allow domain experts to develop systems
  - Express solutions in the problem domain
- Run time configuration of complex systems
  - Emacs, Stendhal
- Simplified recipe scripts
  - Build systems, installers
- Connect two or more different languages or services
  - ColdFusion Markup Language
- Generate models and services in multiple languages
  - One source multiple targets (e.g. Ruby on Rails)

DSL implementation styles

- Internal (embedded)
  - Uses the syntax of a host language
    - Already have a parser and run time
  - Easier in flexible, low ceremony languages
    - Ruby, Groovy, Lisp

- External
  - Custom syntax
    - Need to write code to parse and run it
  - Typical example: XML-style configuration files

DSL implementation styles

- Interpreted
  - Read the DSL script and execute it at run time
    - More natural in interpreted languages
  - Good for configuration at run time

- Code generation
  - Read the DSL script and generate code
    - In a general purpose language
      - Often C, C++, Java
    - May then also need to be compiled
  - Not suitable for run time configuration tasks

The exam

- Online via Blackboard
- Duration: 60 minutes
- 30 multiple choice questions
  - Covering the entire course
    - Lab coursework
    - Workshops
    - Reading material in Moodle

Practice exam

- Online via Blackboard
  - Assessment section, bottom of the page
- Duration: 20 minutes
- Ten multiple choice questions
  - Representative of the real thing

SO, IS THIS GOING TO BE ON THE TEST??
HOW TO ANSWER:
IF I SAY YES
IF I SAY NO
THAT'S TOO EASY.
THEY WON'T LEARN IT.
BUT THEN IF IT'S NOT ON THE TEST
IF I SAY MAYBE OR ANSWER VAGUELY
THEY'LL COMPLAIN
IT WASN'T GOING TO BE, BUT THANKS FOR THE SUGGESTION...
NOOOO!!
heh heh.
JORGE CHAM © 2016
WWW.PHDCOMICS.COM

Next week

- Your final coursework deadline is **this week**
  - Friday 1700.

- The Developer Showcases will happen during the workshop slots.