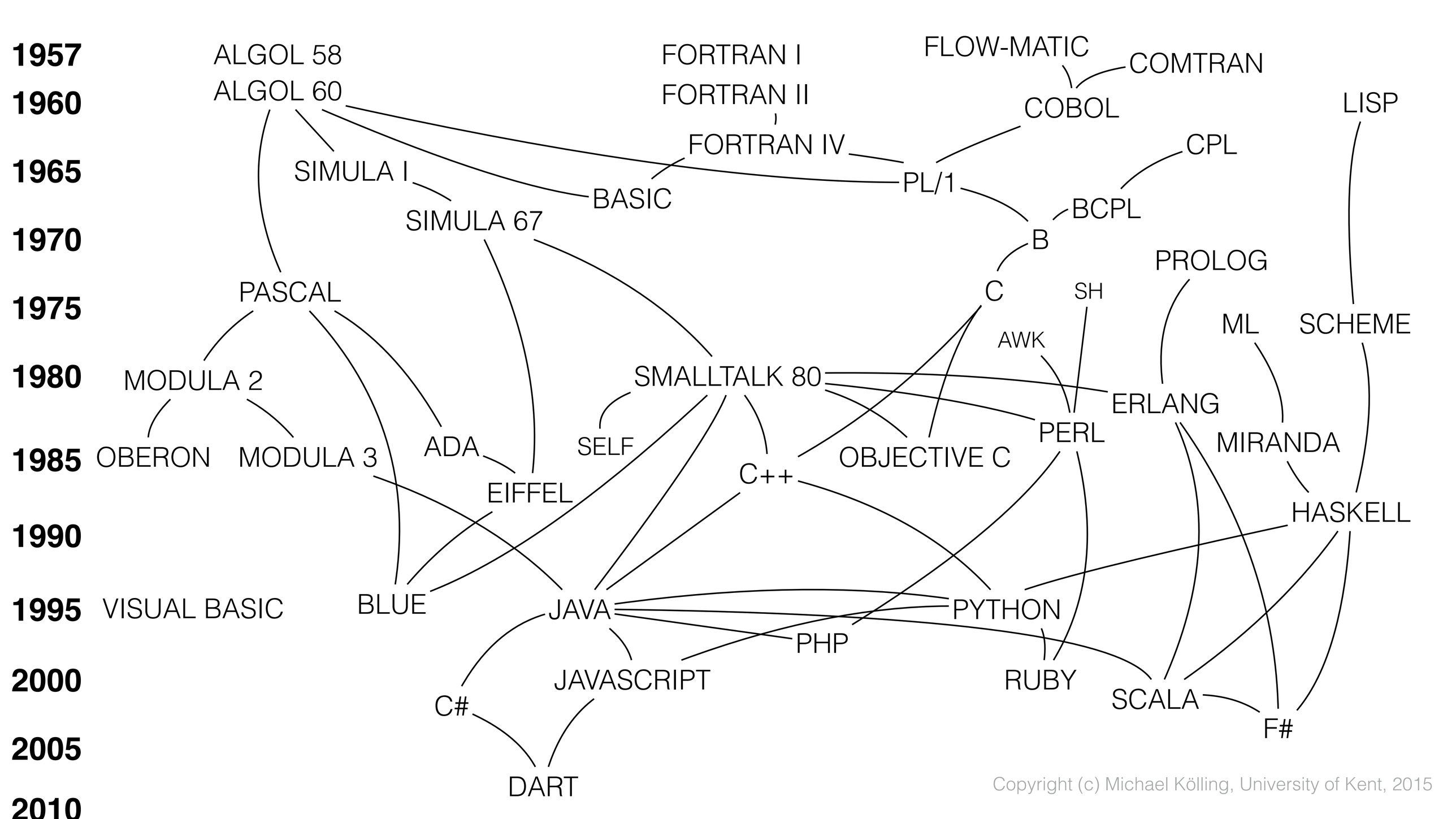
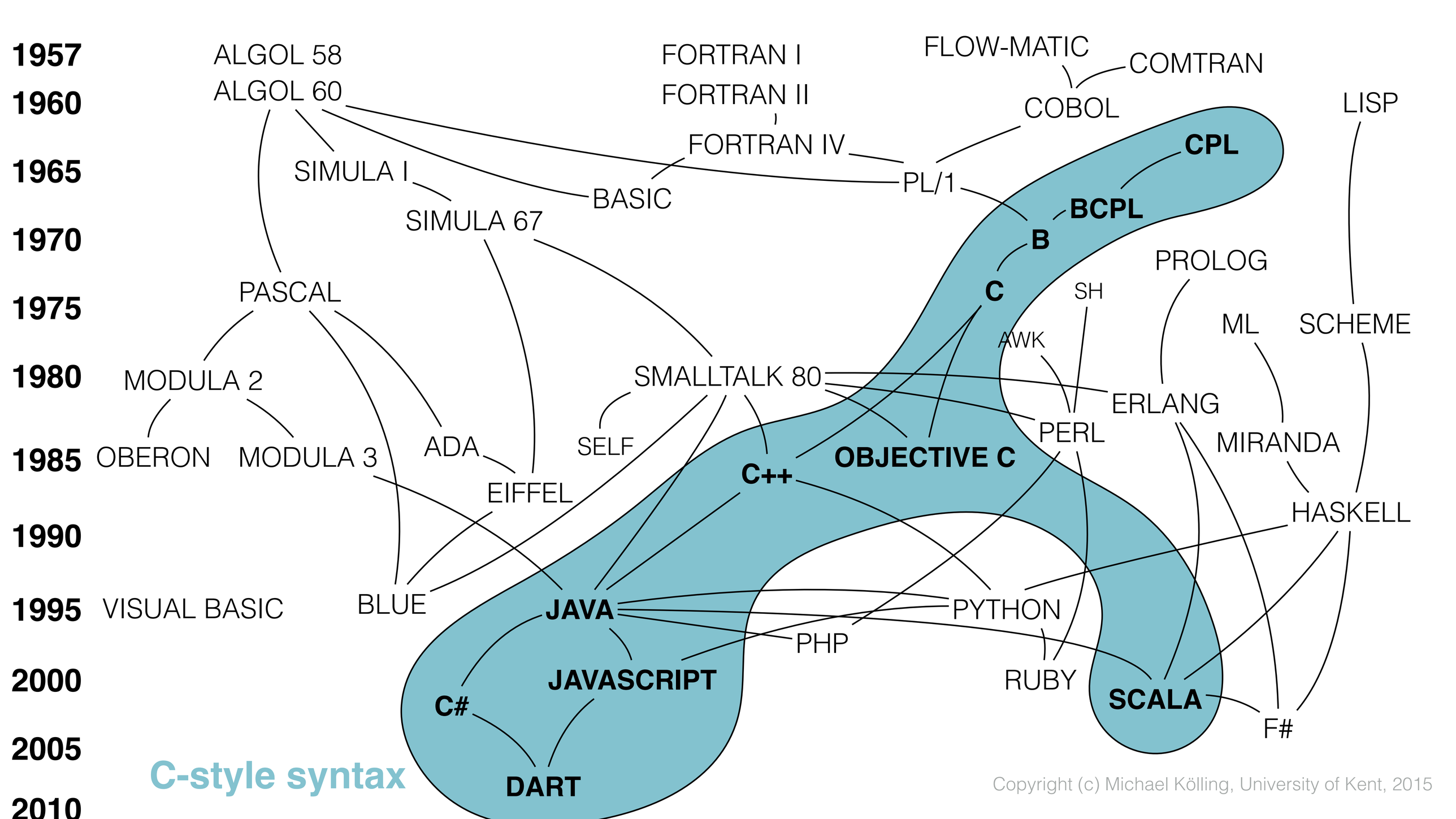
What is OOP
(and why does it matter?)

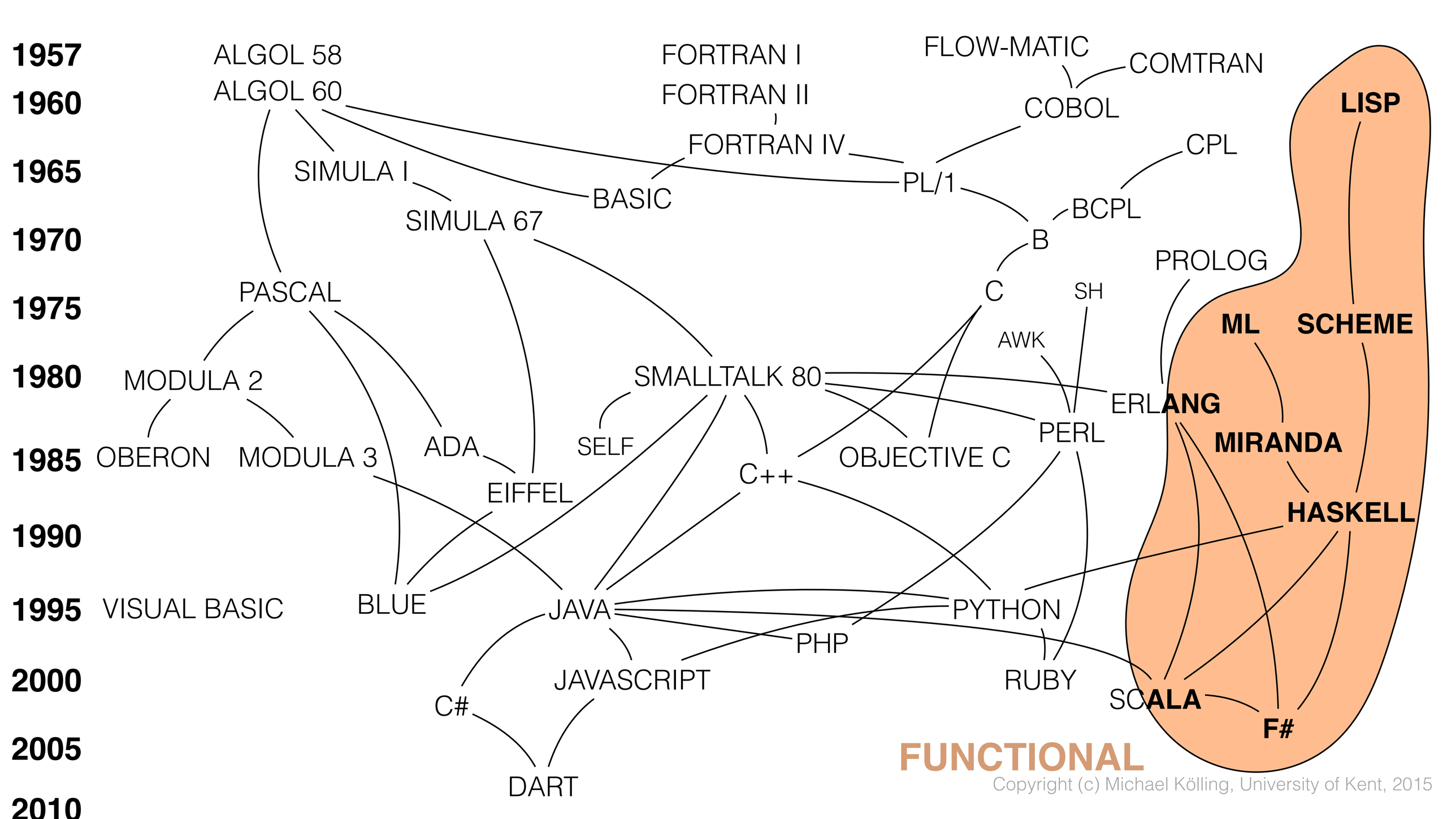
CAS conference 2015

Michael Kölling
University of Kent

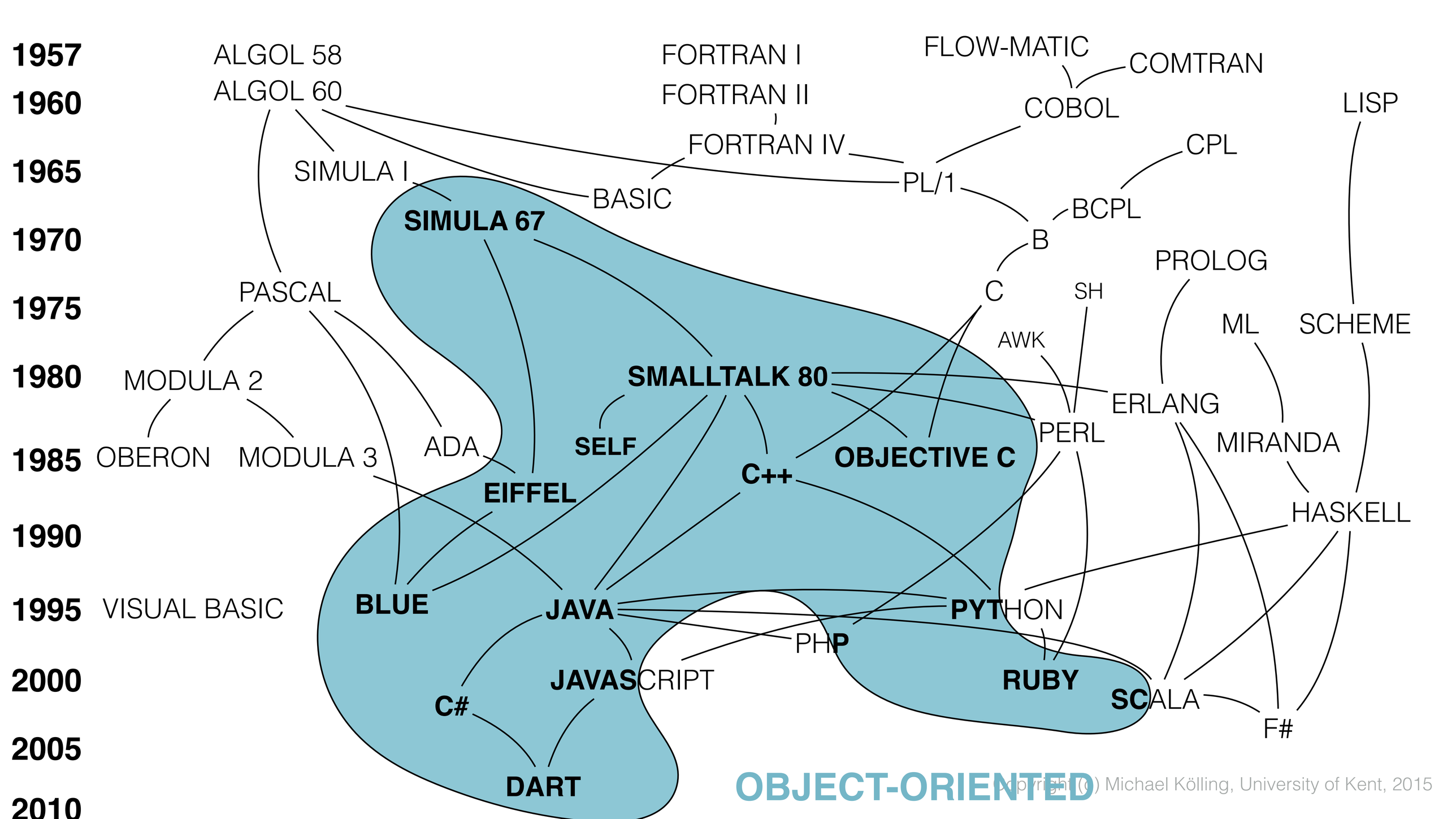


Copyright (c) Michael Kölling, University of Kent, 2015

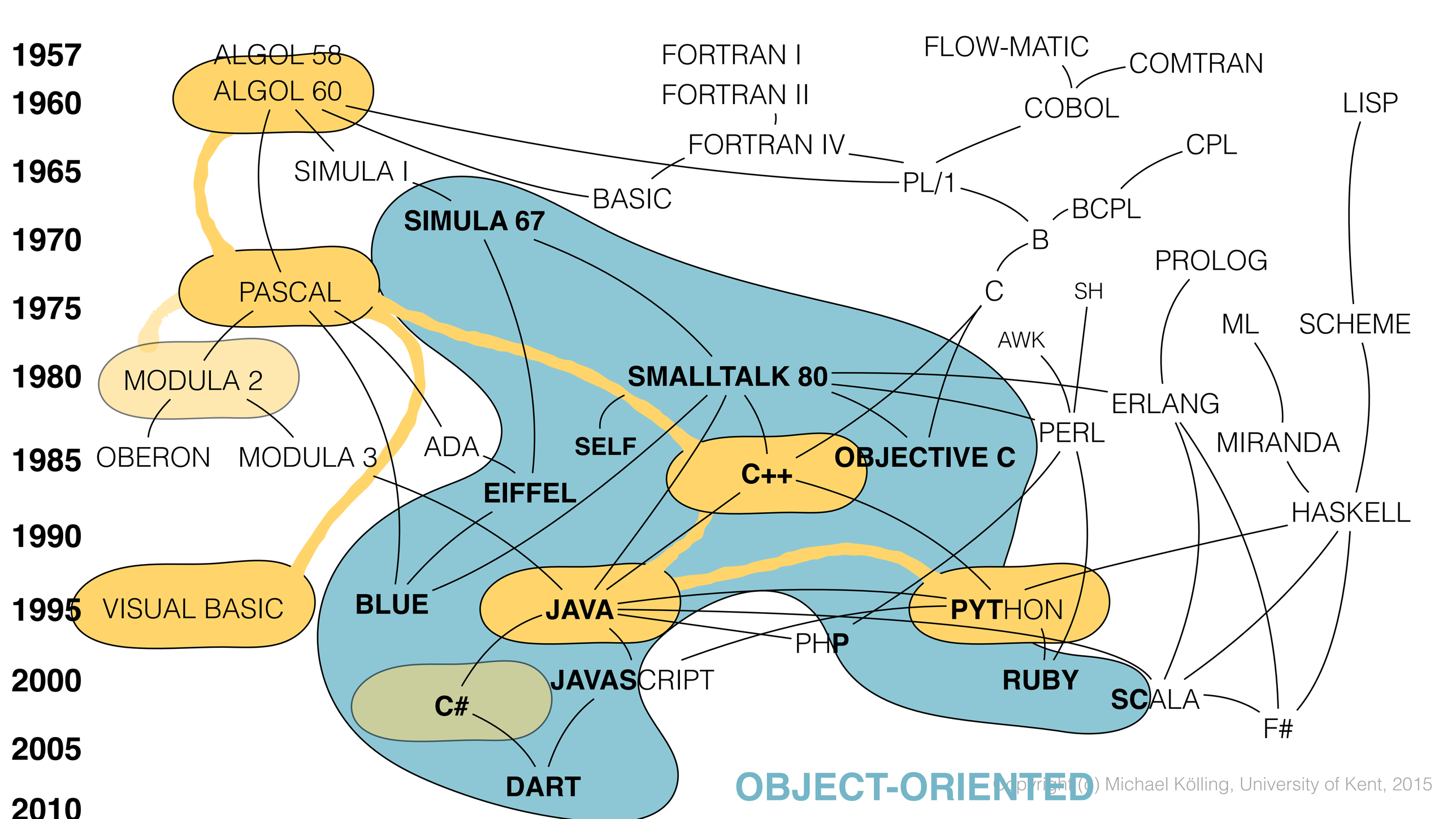

1957
1960
1965
1970
1975
1980
1985
1990
1995
2000
2005
2010
ALGOL 58
ALGOL 60
FORTRAN I
FORTRAN II
FORTRAN IV
FLOW-MATIC
COMTRAN
COBOL
LISP
SIMULA I
SIMULA 67
BASIC
PL/1
CPL
BCPL
B
C
PASCAL
PROLOG
SH
AWK
ML
SCHEME
MODULA 2
SMALLTALK 80
ERLANG
PERL
OBERON
MODULA 3
ADA
SELF
EIFFEL
OBJECTIVE C
C++
MIRANDA
HASKELL
VISUAL BASIC
BLUE
JAVA
PYTHON
PHP
JAVASCRIPT
RUBY
SCALA
C#
F#
DART
C-style syntax
Copyright (c) Michael Kölling, University of Kent, 2015



Copyright (c) Michael Kölling, University of Kent, 2015

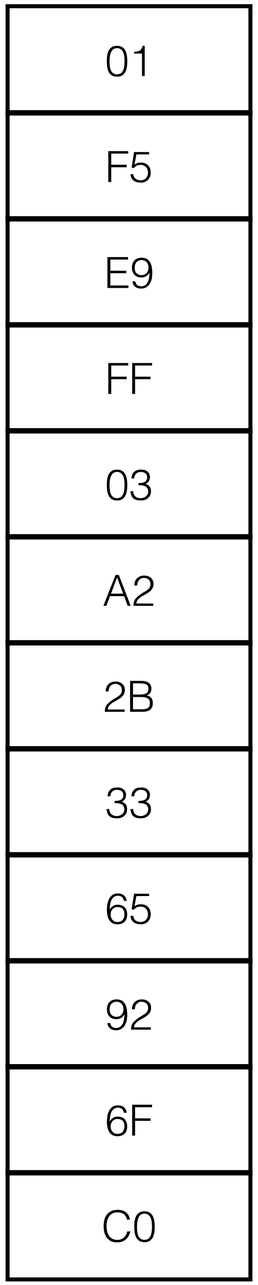


Copyright (c) Michael Kölling, University of Kent, 2015



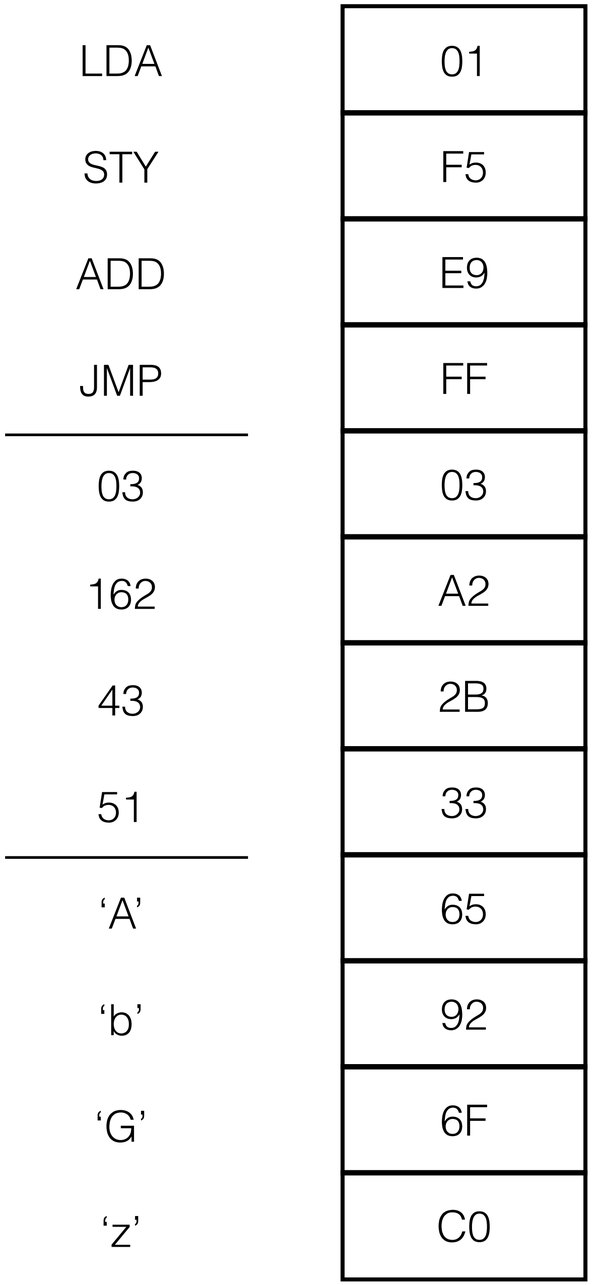
Copyright (c) Michael Kölling, University of Kent, 2015

Machine code



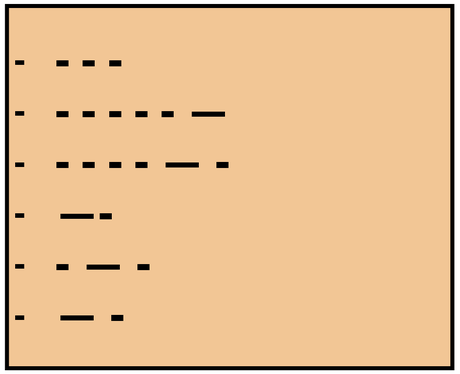
Copyright (c) Michael Kölling, University of Kent, 2015

Assembler

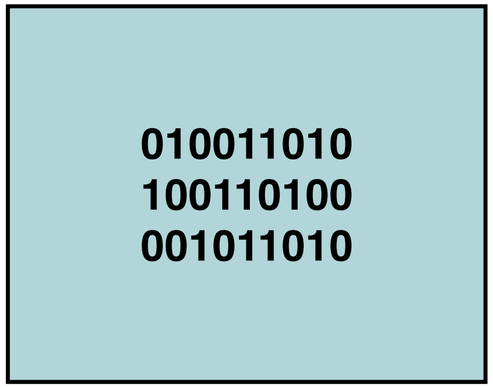


Copyright (c) Michael Kölling, University of Kent, 2015

Higher level programming



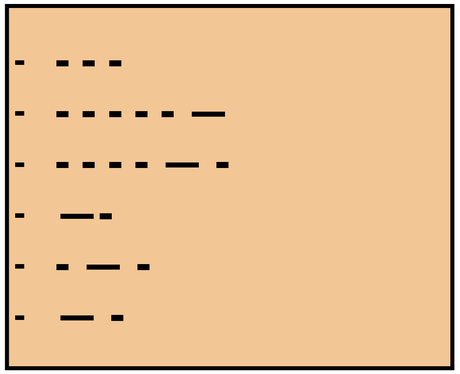
code



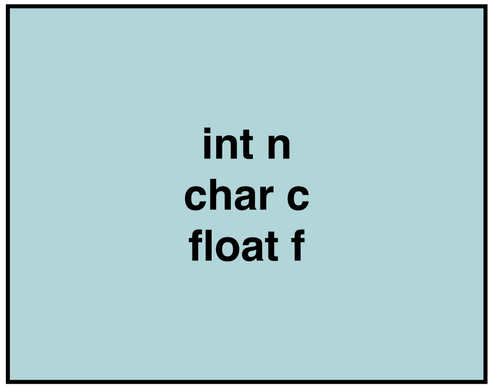
data

Copyright (c) Michael Kölling, University of Kent, 2015

Higher level programming – with types



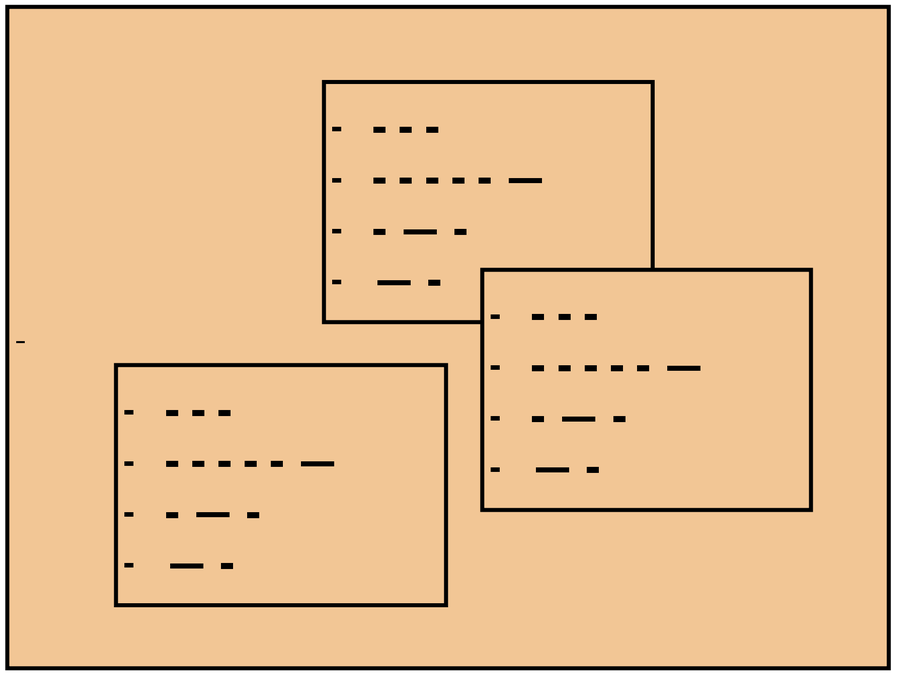
code



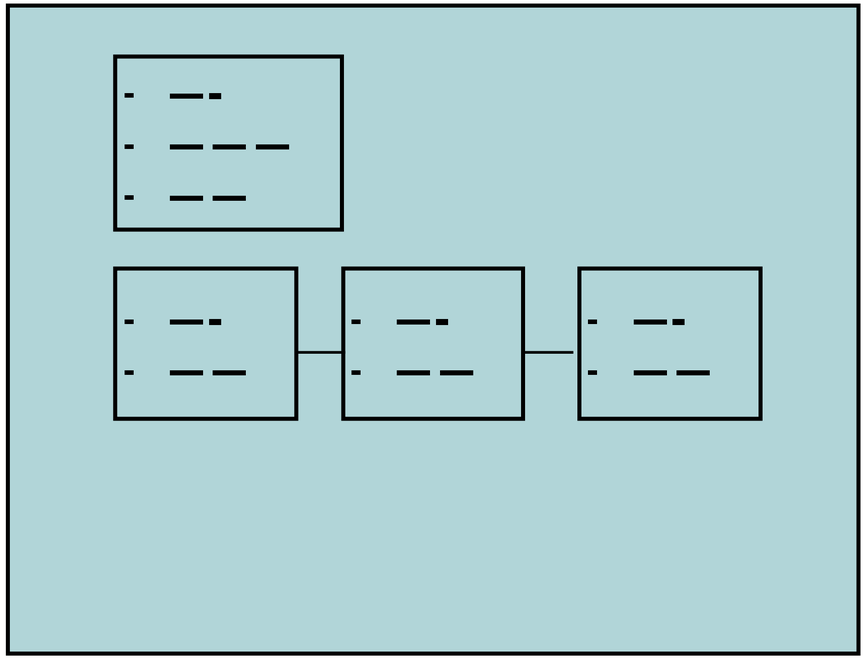
data

Copyright (c) Michael Kölling, University of Kent, 2015

Structured programming



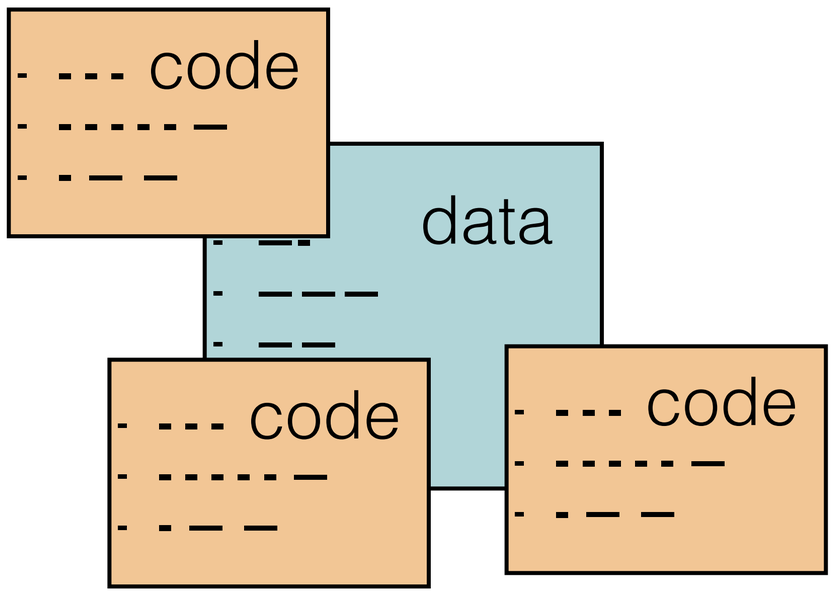
code

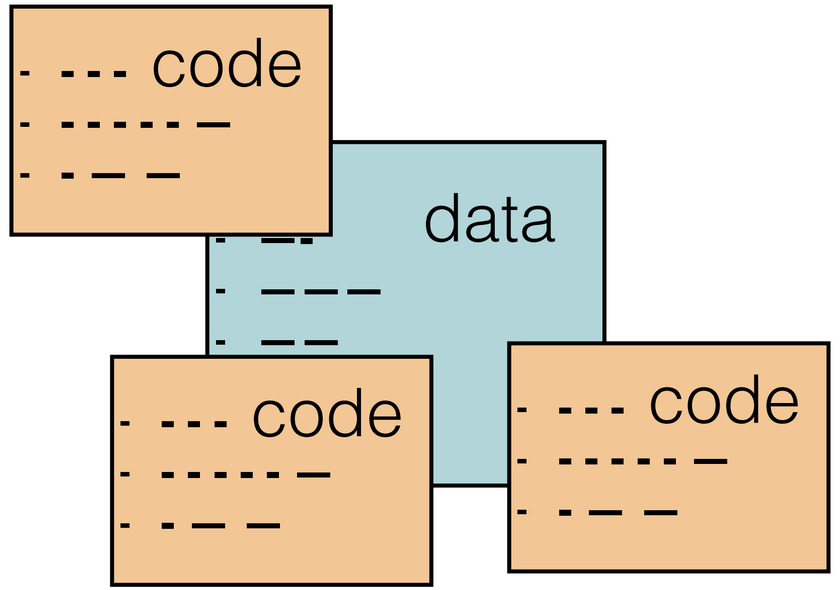


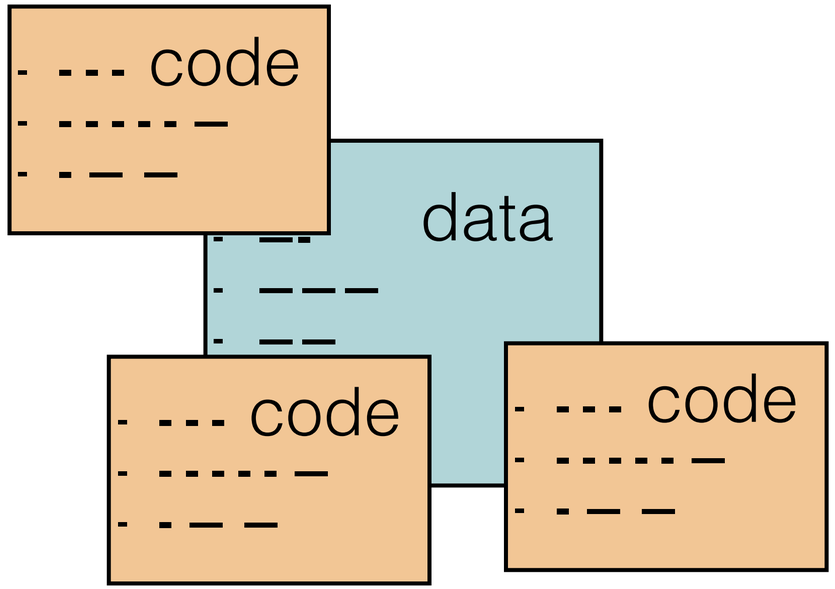
data

Copyright (c) Michael Kölling, University of Kent, 2015

Object-oriented programming

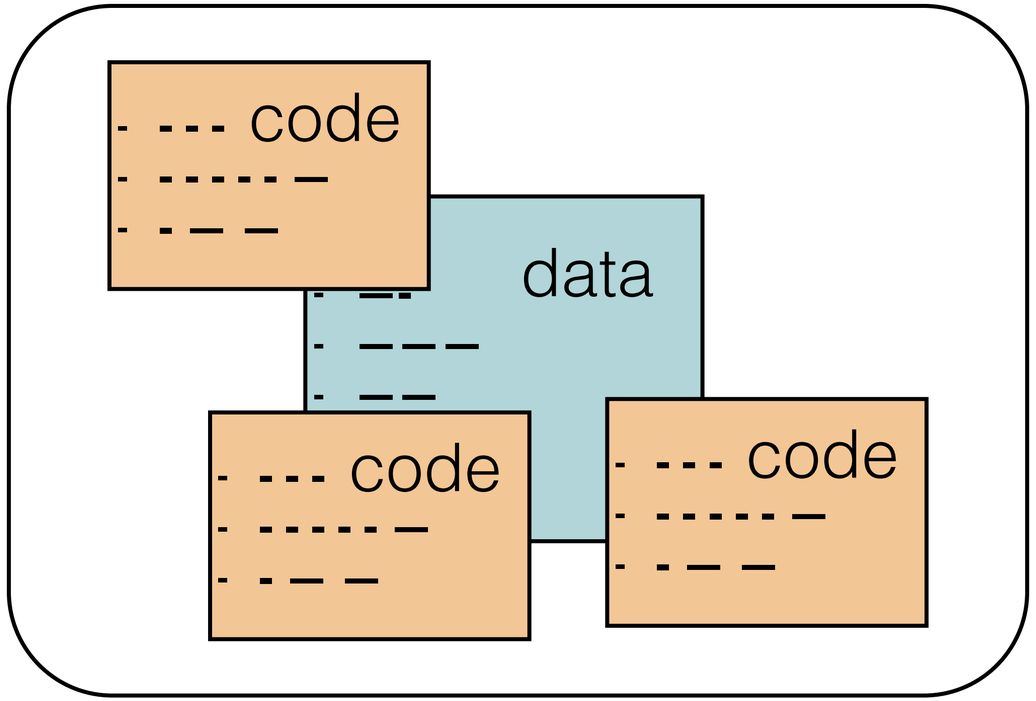






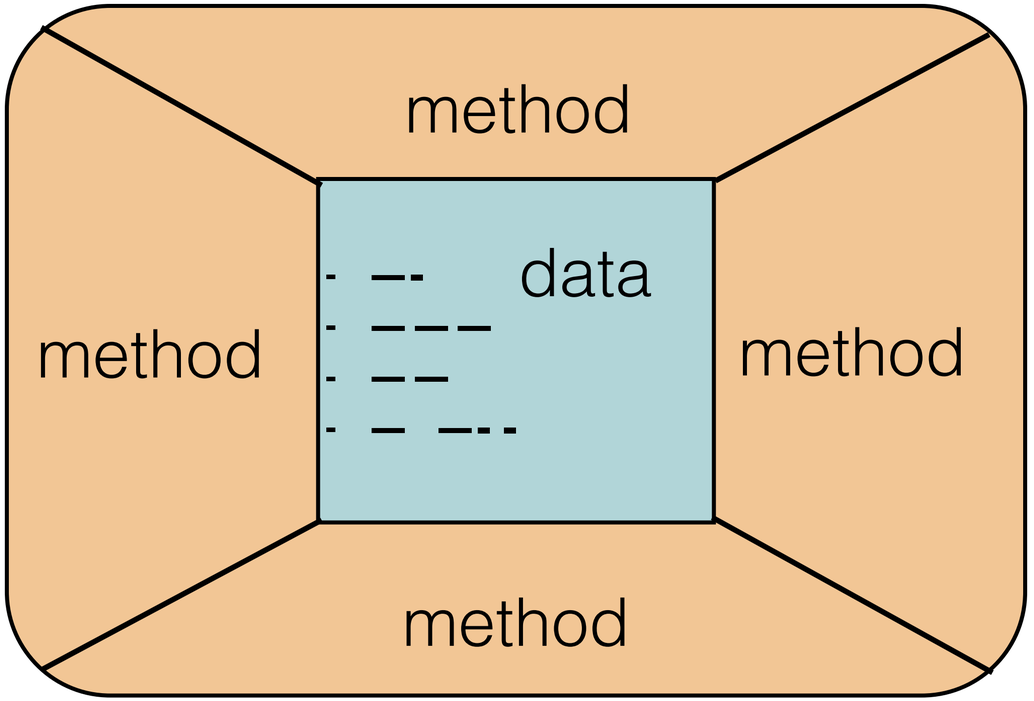
Copyright (c) Michael Kölling, University of Kent, 2015

Object-oriented programming



Copyright (c) Michael Kölling, University of Kent, 2015

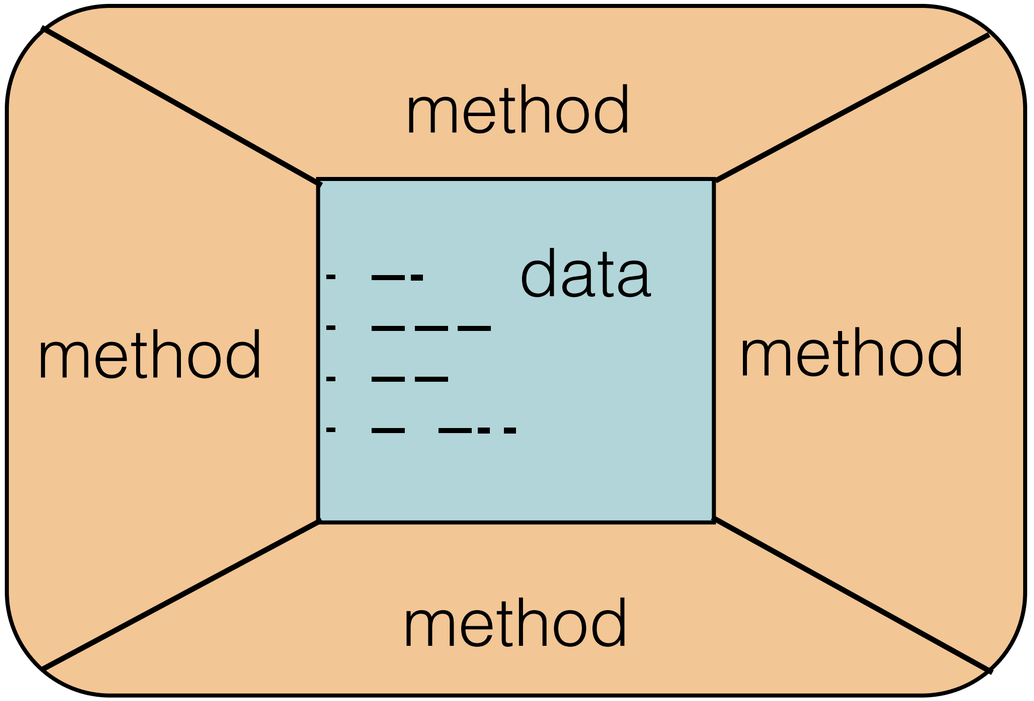
Object-oriented programming



Encapsulation
(Information hiding)

Copyright (c) Michael Kölling, University of Kent, 2015

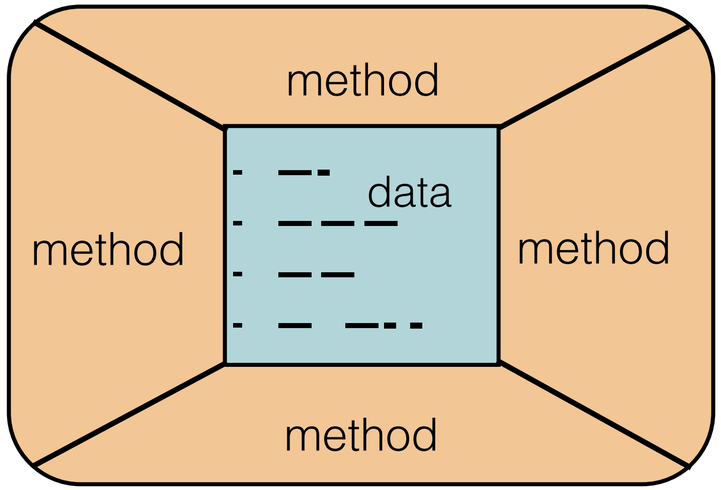
Object-oriented programming



Encapsulation
(Information hiding)

Copyright (c) Michael Kölling, University of Kent, 2015

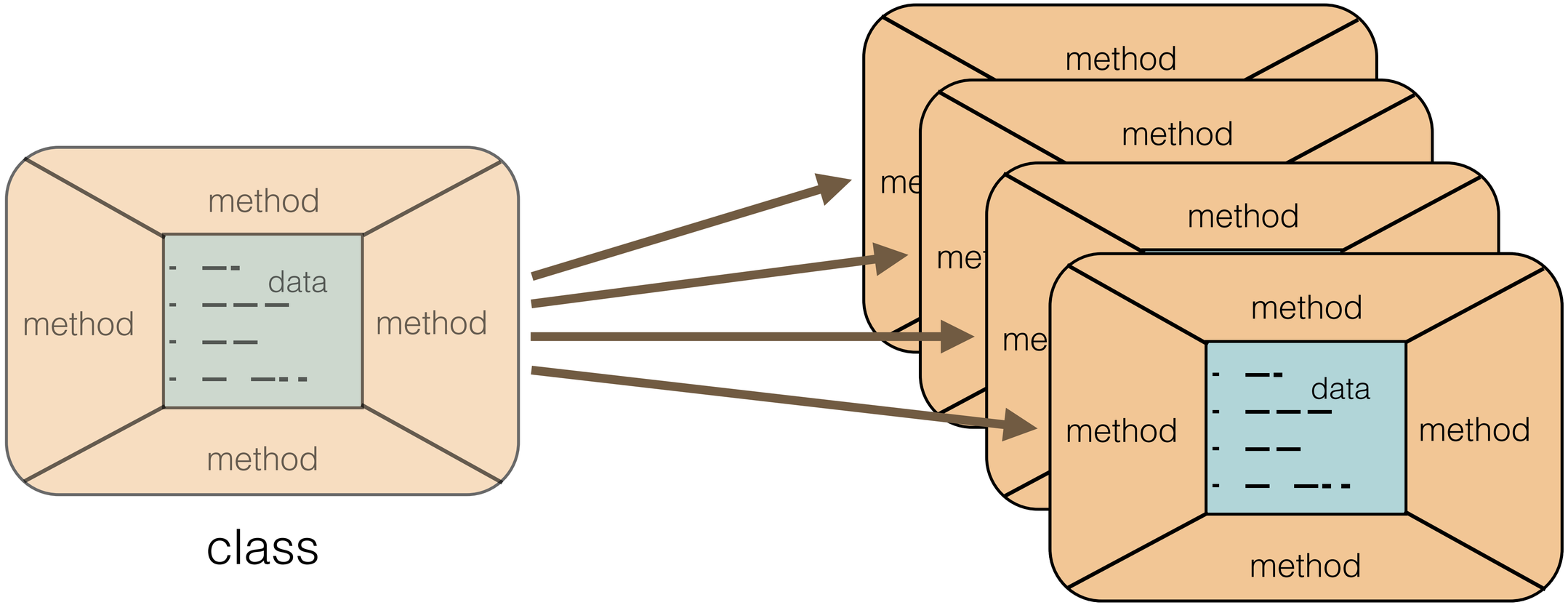
Object-oriented programming



class

Copyright (c) Michael Kölling, University of Kent, 2015

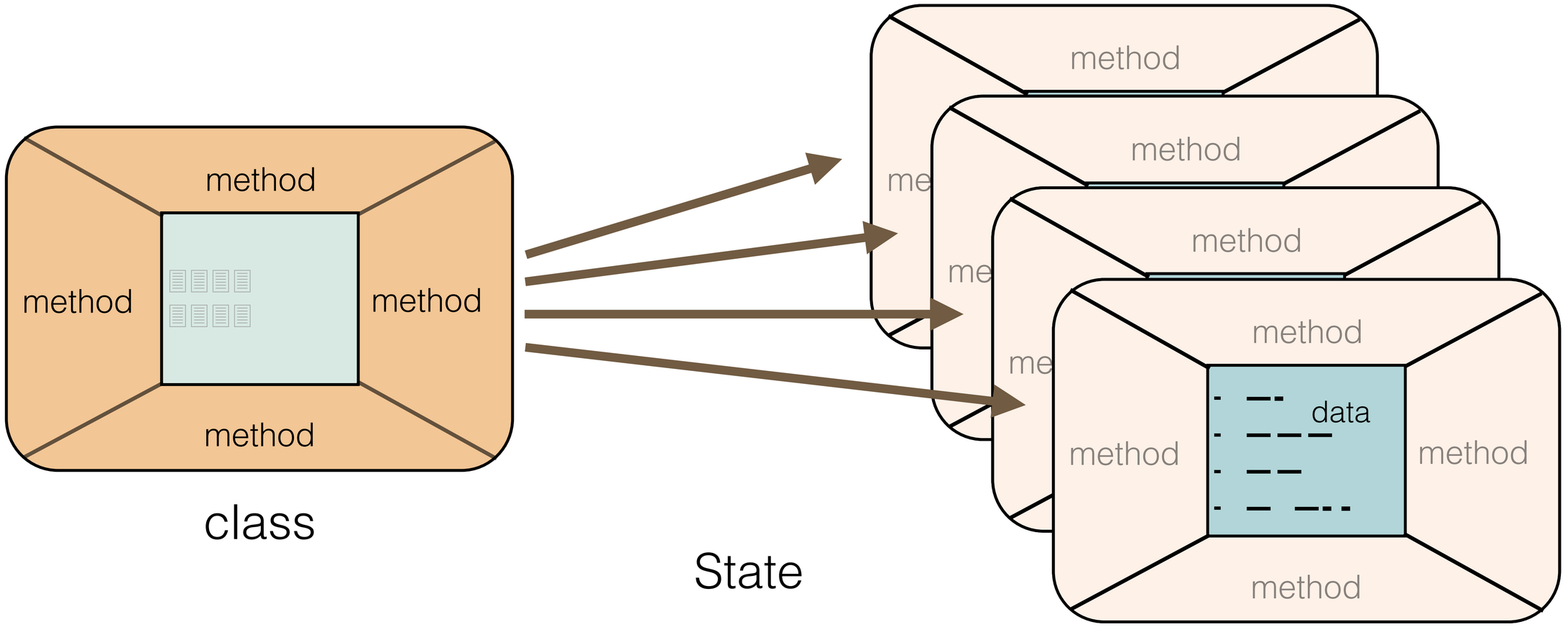
Object-oriented programming



Instantiation

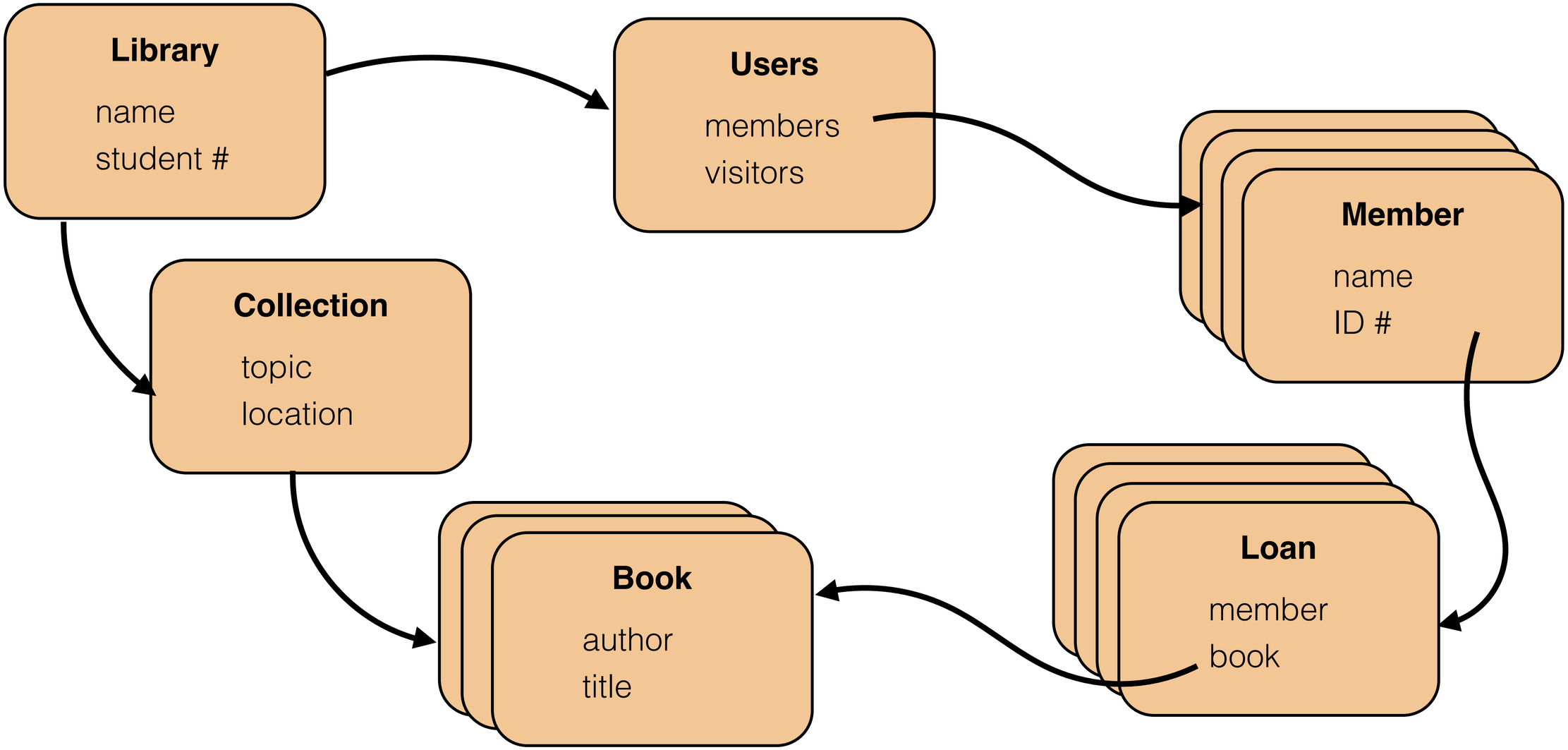
Copyright (c) Michael Kölling, University of Kent, 2015

Object-oriented programming



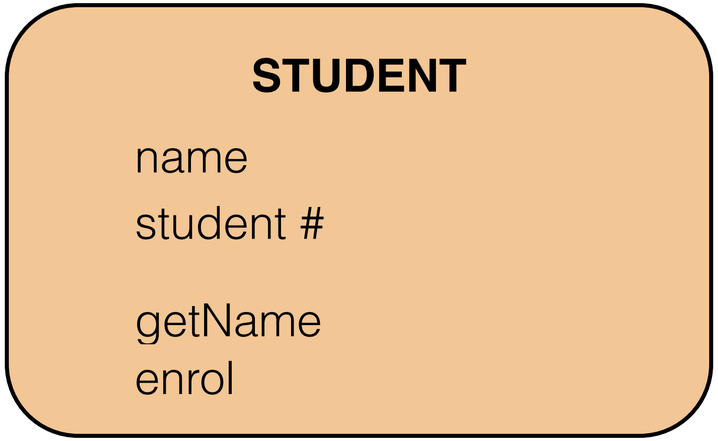
Copyright (c) Michael Kölling, University of Kent, 2015

Object-oriented programming



Copyright (c) Michael Kölling, University of Kent, 2015

Object-oriented programming

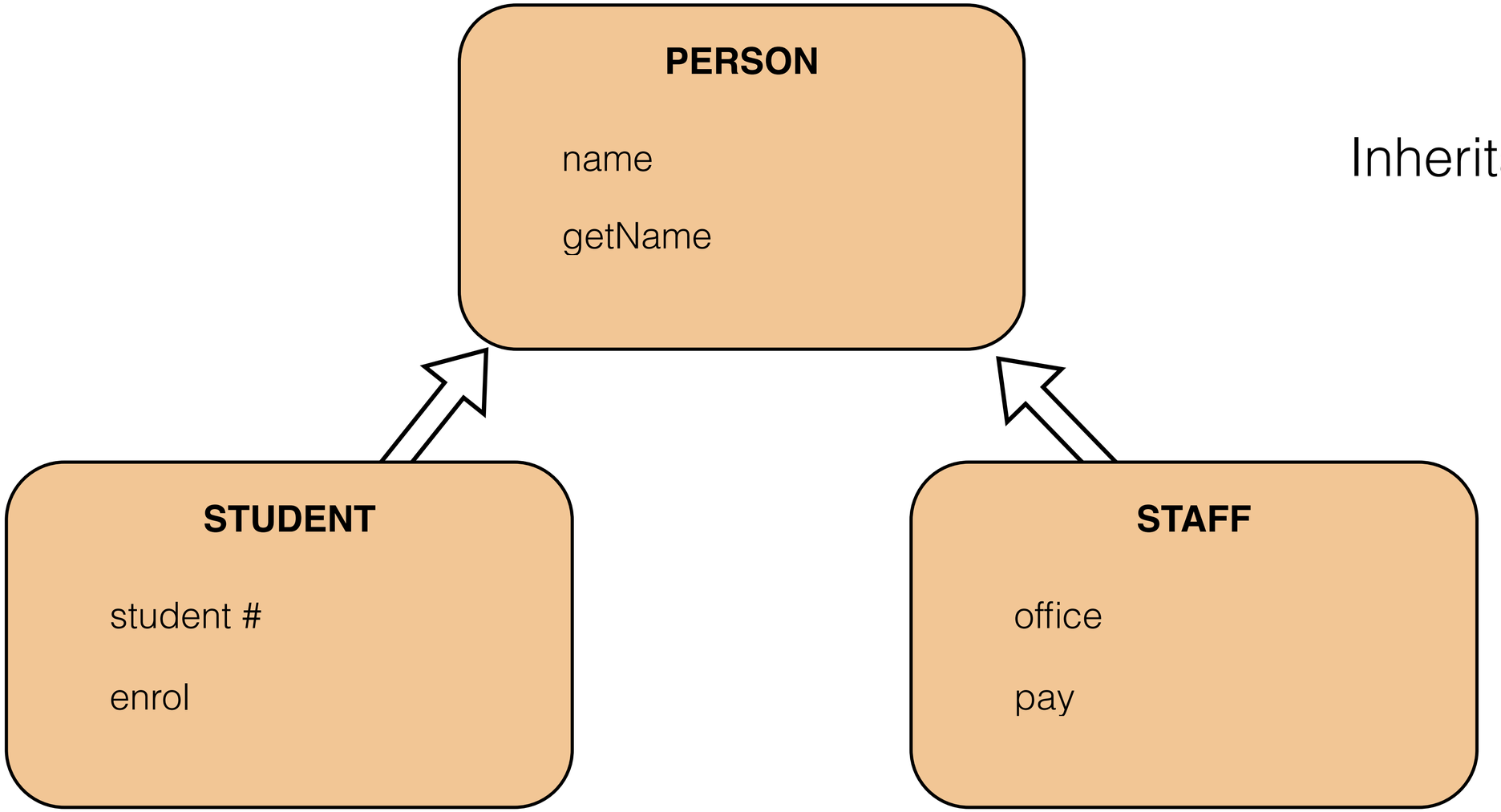


Copyright (c) Michael Kölling, University of Kent, 2015

Object-oriented programming

Copyright (c) Michael Kölling, University of Kent, 2015

Object-oriented programming



Inheritance

Copyright (c) Michael Kölling, University of Kent, 2015

Modularisation

Readability

Maintainability

Extendability

Correctness

Modelling

Copyright (c) Michael Kölling, University of Kent, 2015

PRO:

Modularisation

Readability

Maintainability

Extendability

Correctness

Modelling

CON:

Boilerplate code

Strictness / inflexibility

Copyright (c) Michael Kölling, University of Kent, 2015

And now: How do I teach this?

Copyright (c) Michael Kölling, University of Kent, 2015