

Rubinius

Rubini us

Rubini.us

rubini.us

<http://rubini.us>

Rubinius

<http://godfat.org/slide/2008-12-21-rubinius.pdf>

History and Design Goals

Architecture and Object Model

History and Design Goals

Architecture and Object Model

Evan Phoenix

February of
2006

RubySpec

MSpec

Engine Yard

C VM Shotgun

~~C VM Shotgun~~



C++ VM

CxxTest

LLVM

History and Design Goals

Architecture and Object Model

Reliable, Rock Solid Code

Reliable, Rock Solid Code

Full Test Coverage

健康

Clean, Readable Code

Clean, Readable Code

Little Lines in Each File

Clean, Readable Code

Macro, Code Generator, Rake Task

Clean, Readable Code

CMake

Clean, Readable Code

~~CMake~~

Clean, Readable Code

C++ Object to Ruby Object 1 to 1 Mapping

清新

健康

清新

Modern Techniques

Modern Techniques

Pluggable Garbage Collectors

Modern Techniques

Pluggable Garbage Collectors

- Stop-and-Copy

Modern Techniques

Pluggable Garbage Collectors

- Stop-and-Copy
- Mark-and-Sweep

Modern Techniques

Optimizers

Modern Techniques

Git, Rake, LLVM

Squeak

the Smalltalk-80 Implementation

Squeak

Slang

Squeak

- Alan Kay
- Dan Ingalls
- Adele Goldberg

Smalltalk

Xerox PARC

Smalltalk

Object-Oriented

(different from Simula and C++)

Smalltalk

GUI

Smalltalk

MVC

History and Design Goals

Architecture and Object Model

Real Machine



A diagram consisting of two stacked rectangular boxes. The top box is light blue and contains the text 'C++ Virtual Machine'. The bottom box is a darker blue and contains the text 'Real Machine'. Both boxes have a thin dark blue border.

C++ Virtual Machine

Real Machine

kernel/bootstrap

C++ Virtual Machine

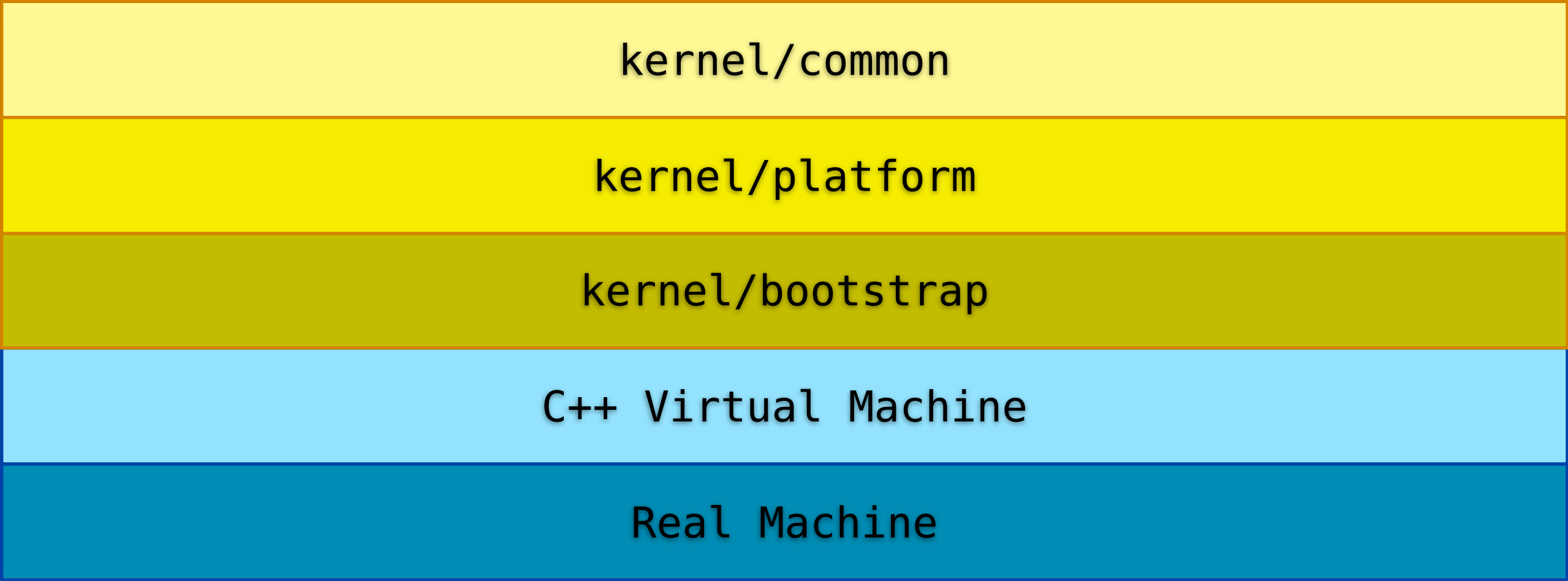
Real Machine

kernel/platform

kernel/bootstrap

C++ Virtual Machine

Real Machine



kernel/common

kernel/platform

kernel/bootstrap

C++ Virtual Machine

Real Machine

kernel/delta

kernel/common

kernel/platform

kernel/bootstrap

C++ Virtual Machine

Real Machine

Ruby
Runtime
(kernel)

kernel/delta

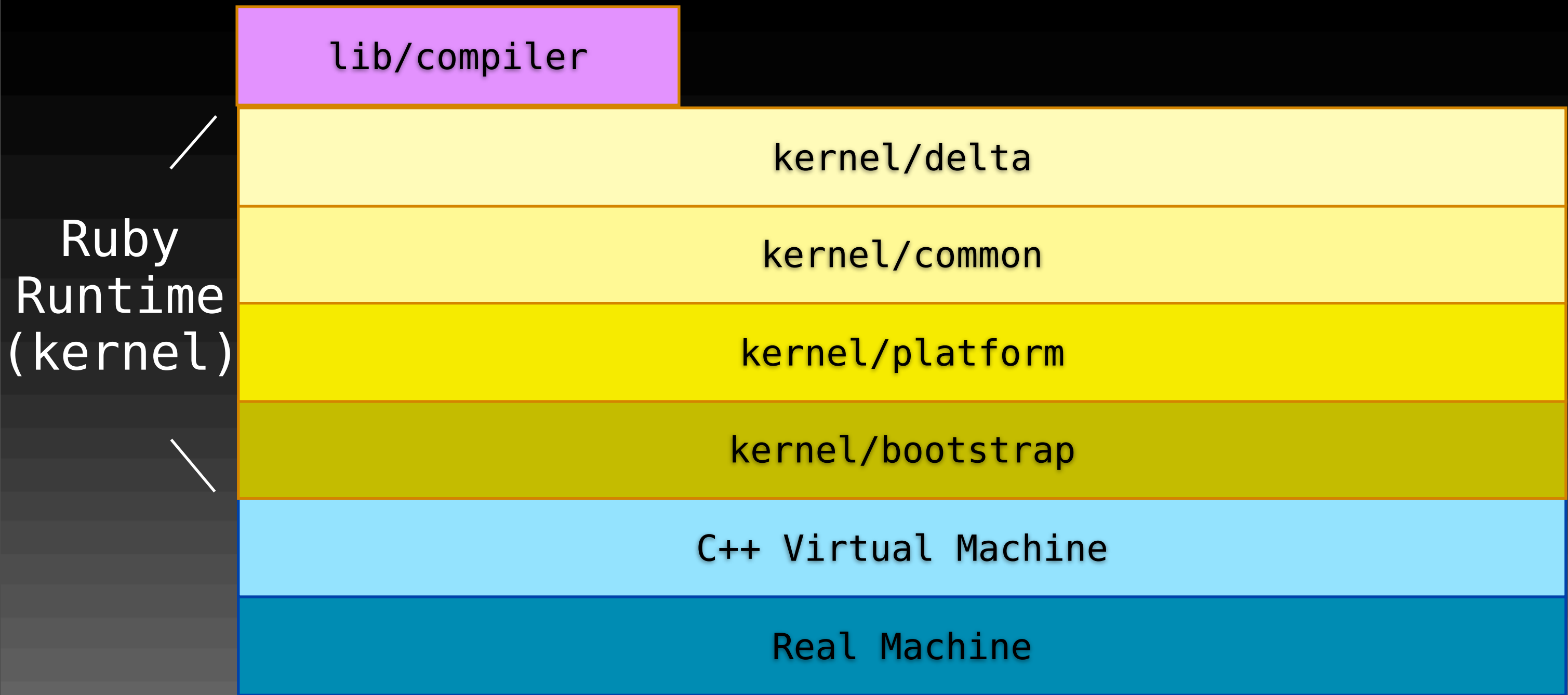
kernel/common

kernel/platform

kernel/bootstrap

C++ Virtual Machine

Real Machine



Ruby
Runtime
(kernel)

lib/compiler

lib/*

kernel/delta

kernel/common

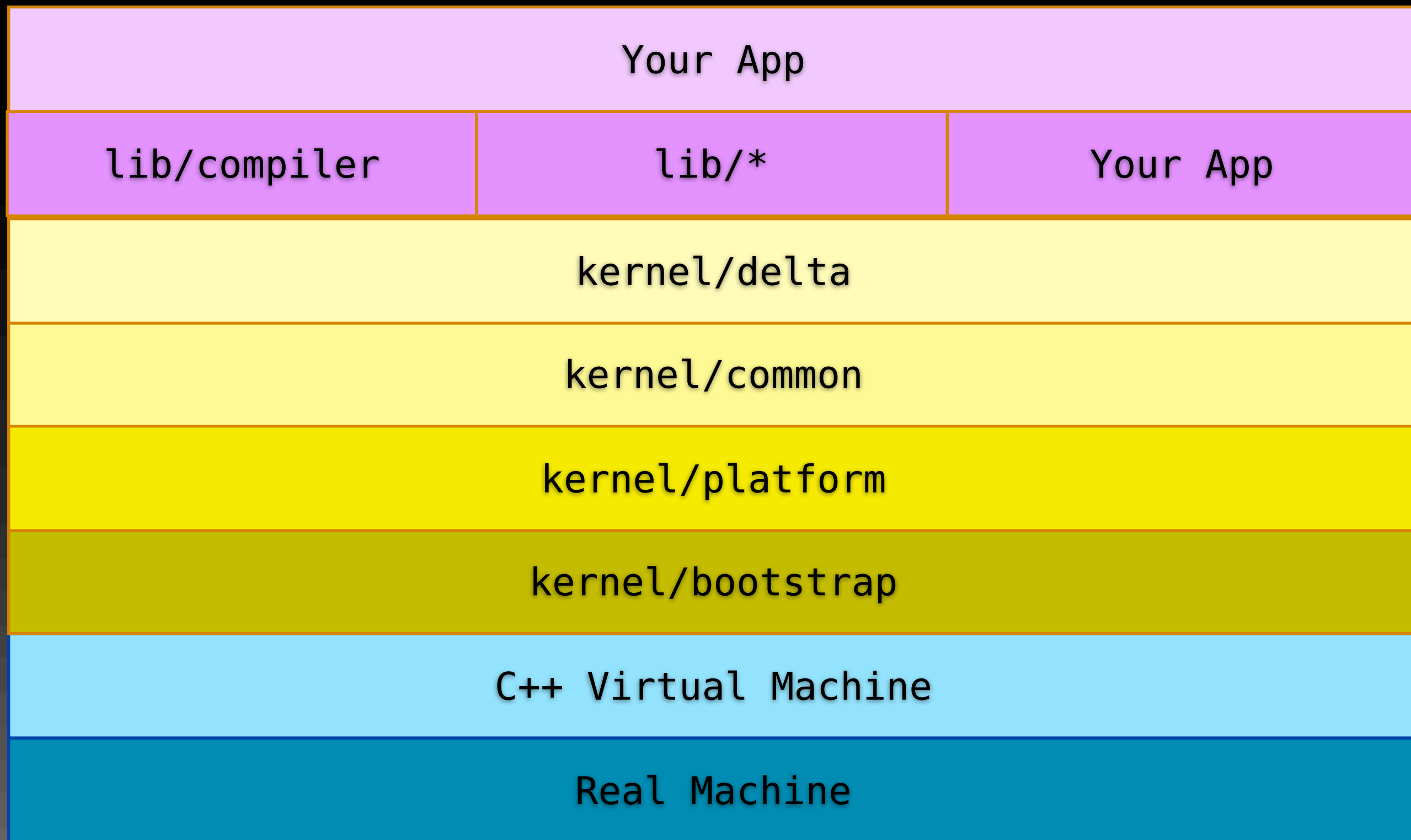
kernel/platform

kernel/bootstrap

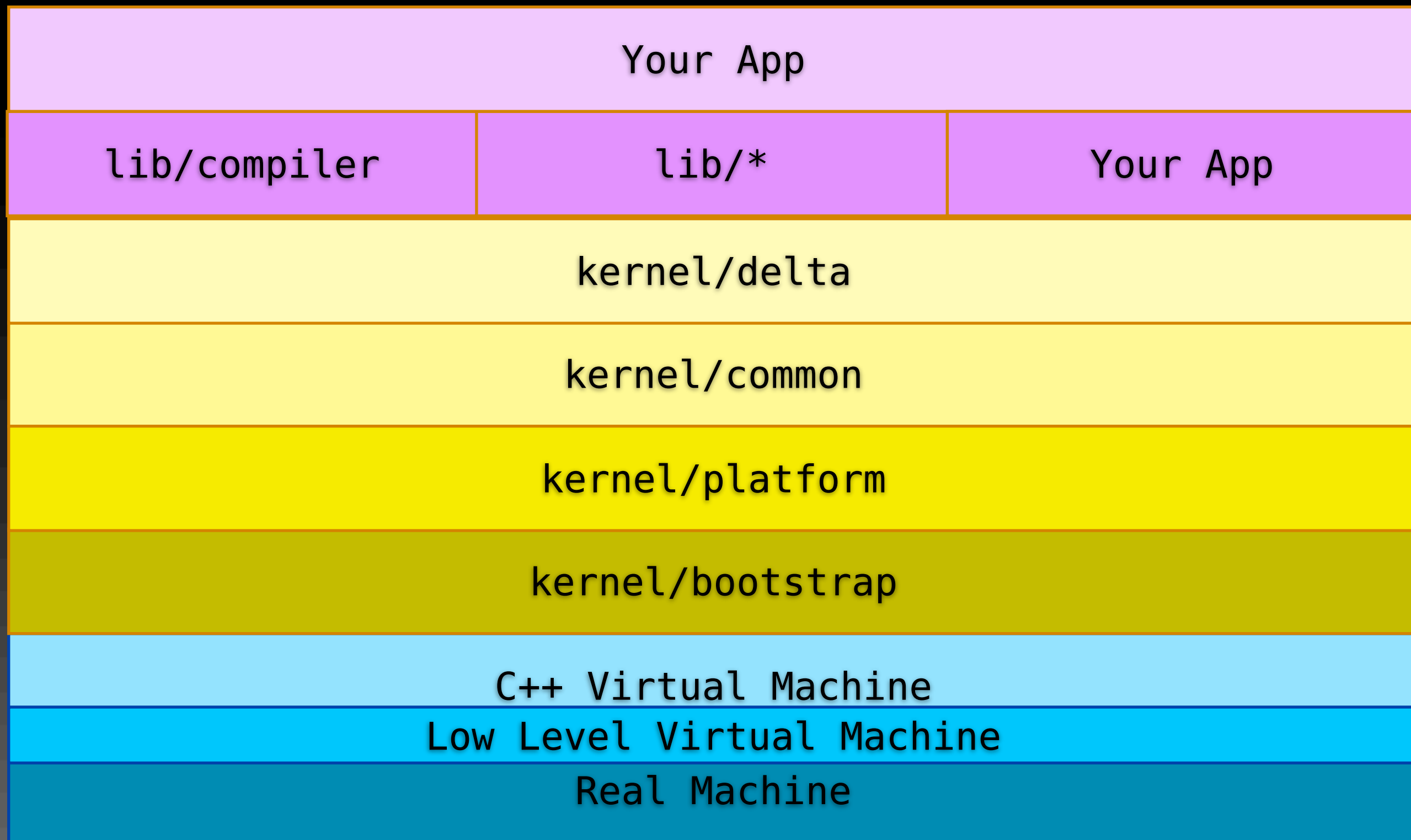
C++ Virtual Machine

Real Machine

Ruby
Runtime
(kernel)



Ruby
Runtime
(kernel)



Outside the Ruby World

Outside the Ruby World

C++ Primitives

Outside the Ruby World

Foreign Function Interface - FFI

Outside the Ruby World

Subtend

History and Design Goals

Architecture and Object Model

Hello

World!

```
def          method_missing m; puts m;          end  
def Object.const_missing c; print "#{c}, "; end
```

Hello.World!

```
puts 'Hello, World!'
```

```
puts 'Hello, World!'
```

```
self.send :puts, 'Hello, World!'
```



```
TOPLEVEL (TOPLEVEL_BINDING.context.receiver ||  
          MethodContext.current.receiver )
```

```
puts 'Hello, World!'  
self.send :puts, 'Hello, World!'
```

```
TOPLEVEL      (TOPLEVEL_BINDING.context.receiver ||  
               MethodContext.current.receiver )
```

```
def __script__  
  
  puts 'Hello, World!'  
  self.send :puts, 'Hello, World!'  
  
end
```

```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__
```

```
  puts 'Hello, World!'
```

```
end
```

```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__
```

```
  puts 'Hello, World!'
```

```
  #<SendSite:0x26 name=puts  
    hits=0 misses=0>
```

```
end
```

```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__  
  print 'Hello,'  
  
  print 'World!'  
  
end
```

CompiledMethod (method(:__script__).compiled_method)

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=0 misses=0>  
  print 'World!'  
  #<SendSite:0x28 name=print  
    hits=0 misses=0>  
end
```

```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__  
  print 'Hello,'  
  
  print 'World!'  
  
end
```

```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=0 misses=0>  
  print 'World!'  
  
end
```


CompiledMethod (method(:__script__).compiled_method)

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=0 misses=0>  
  print 'World!'  
  #<SendSite:0x28 name=print  
    hits=0 misses=0>  
end
```

```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__  
  print 'Hello,'  
  
  print 'World!'  
  
end
```

CompiledMethod (method(:__script__).compiled_method)

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=1 misses=0>  
  print 'World!'  
  
end
```

CompiledMethod (method(:__script__).compiled_method)

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=1 misses=0>  
  print 'World!'  
  #<SendSite:0x28 name=print  
    hits=1 misses=0>  
end
```

```
class C  
  def print  
  end  
end
```

```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__  
  print 'Hello,'  
  
  print 'World!'  
  
end
```

```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=0 misses=0>  
  print 'World!'  
  
end
```

CompiledMethod (method(:__script__).compiled_method)

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=0 misses=0>  
  print 'World!'  
  #<SendSite:0x28 name=print  
    hits=0 misses=0>  
end
```



```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__  
  print 'Hello,'  
  
  print 'World!'  
  
end
```

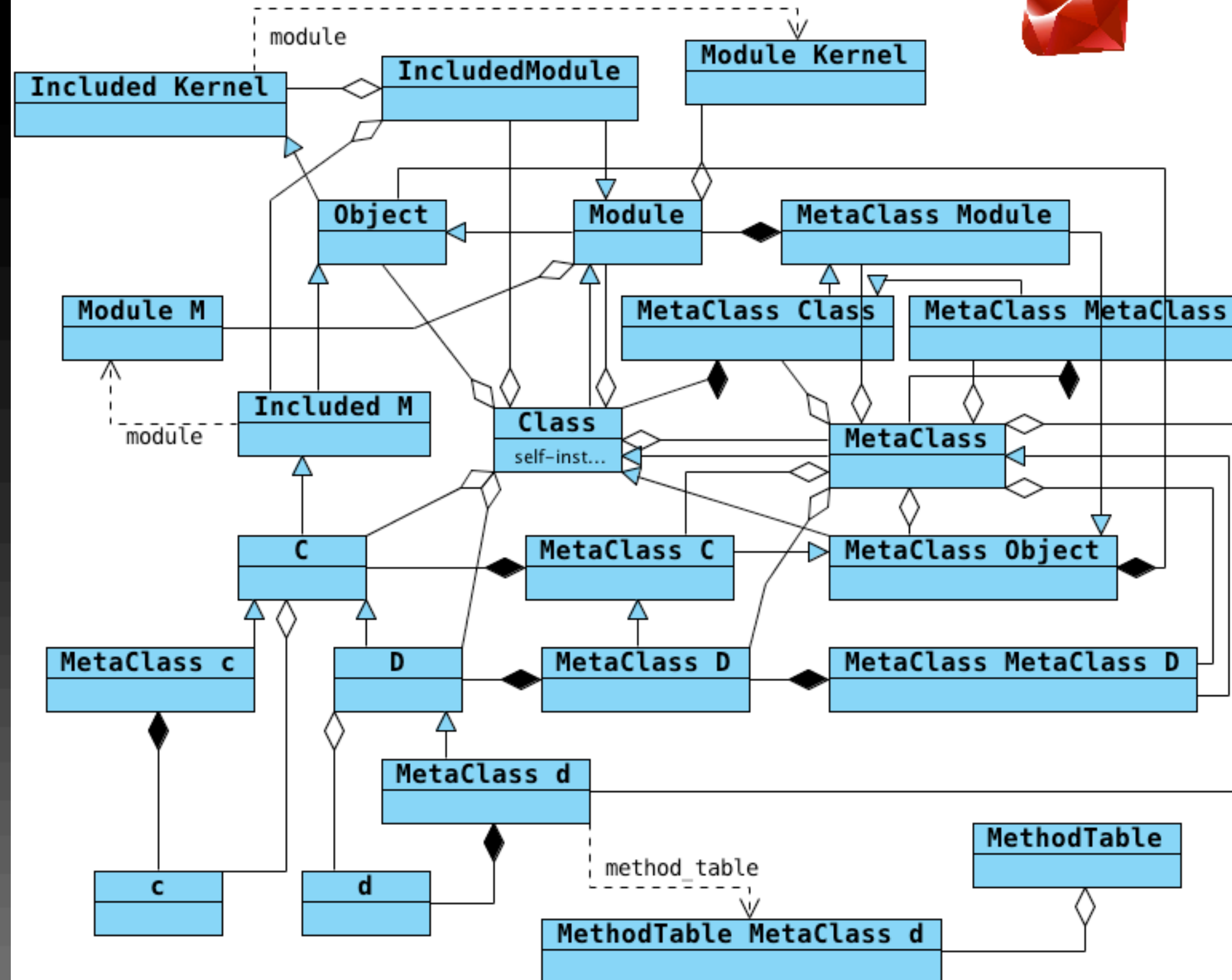
```
CompiledMethod (method(:__script__).compiled_method)
```

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=1 misses=0>  
  print 'World!'  
  
end
```

CompiledMethod (method(:__script__).compiled_method)

```
def __script__  
  print 'Hello,'  
  #<SendSite:0x26 name=print  
    hits=1 misses=0>  
  print 'World!'  
  #<SendSite:0x28 name=print  
    hits=1 misses=0>  
end
```

Ruby Object Model (Labyrinth?)



- CompiledMethod

- MethodContext

- StaticScope

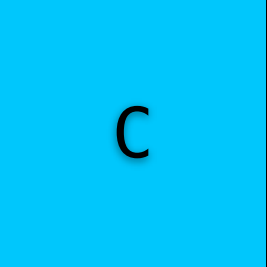
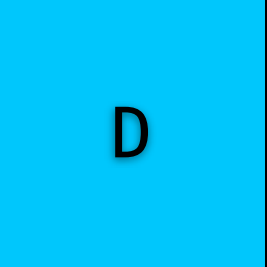
- SendSite

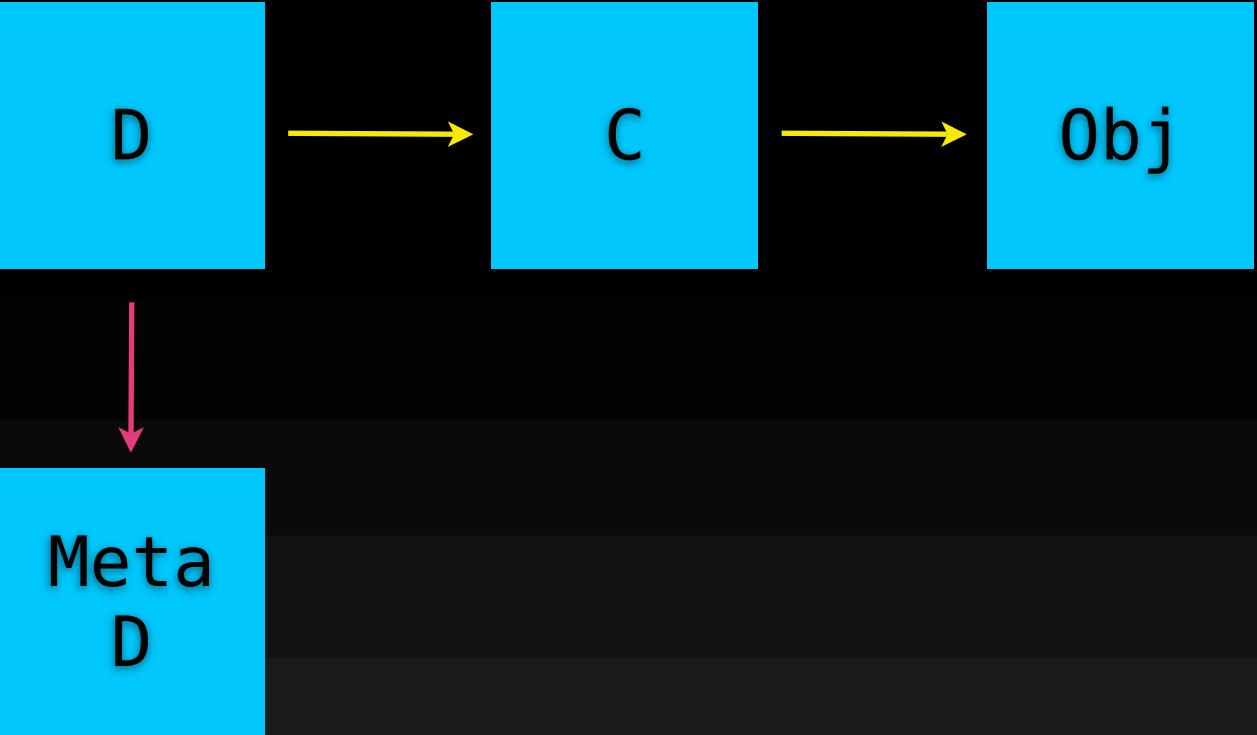
- MethodTable

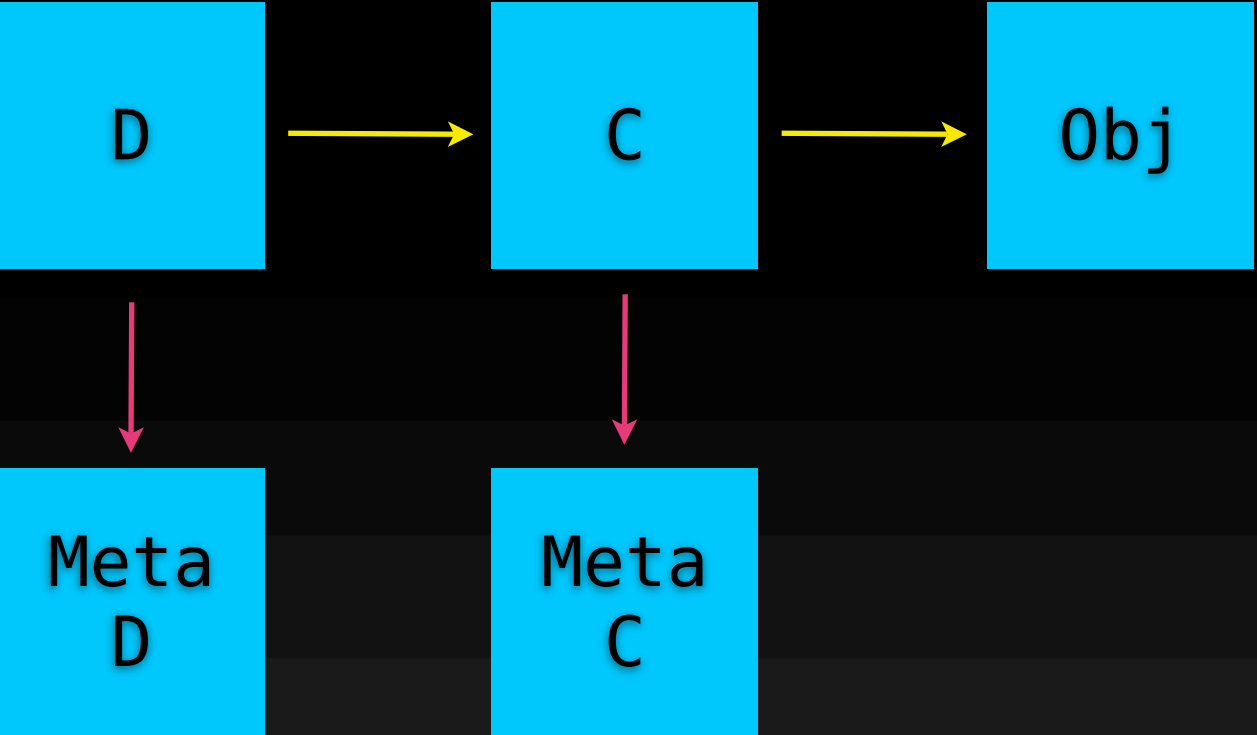
- LookupTable

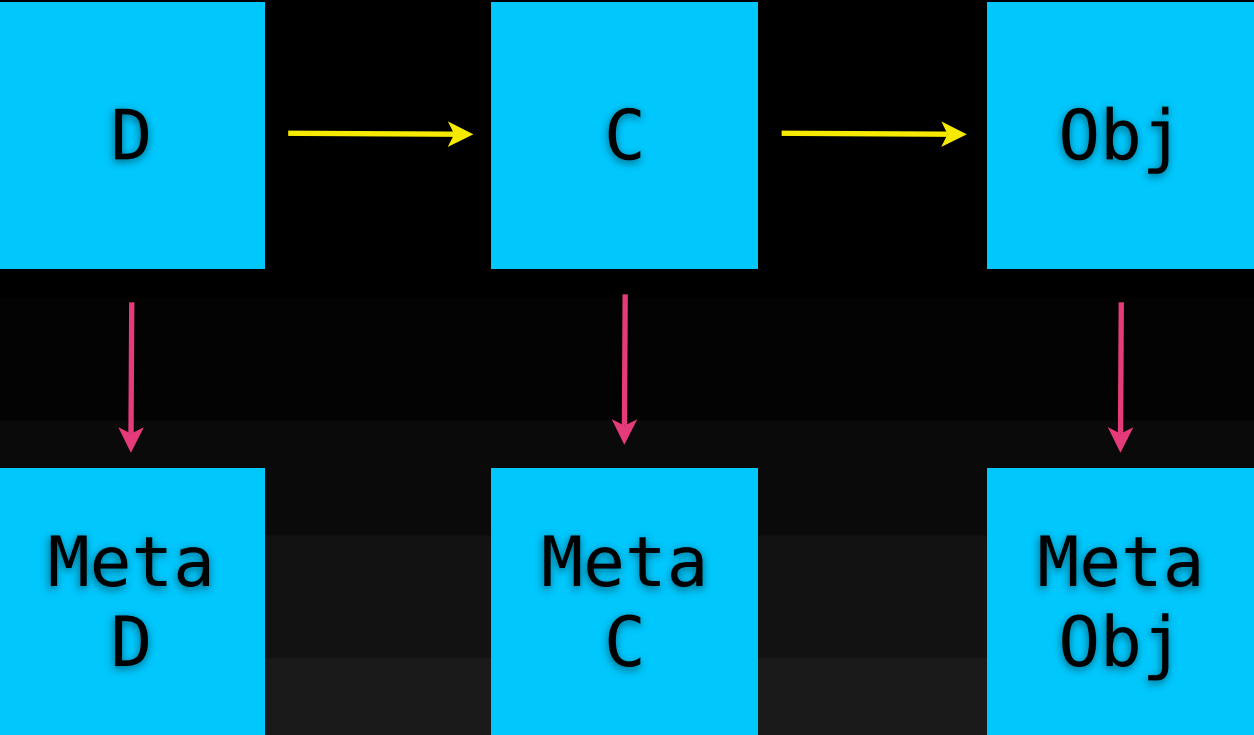
Method

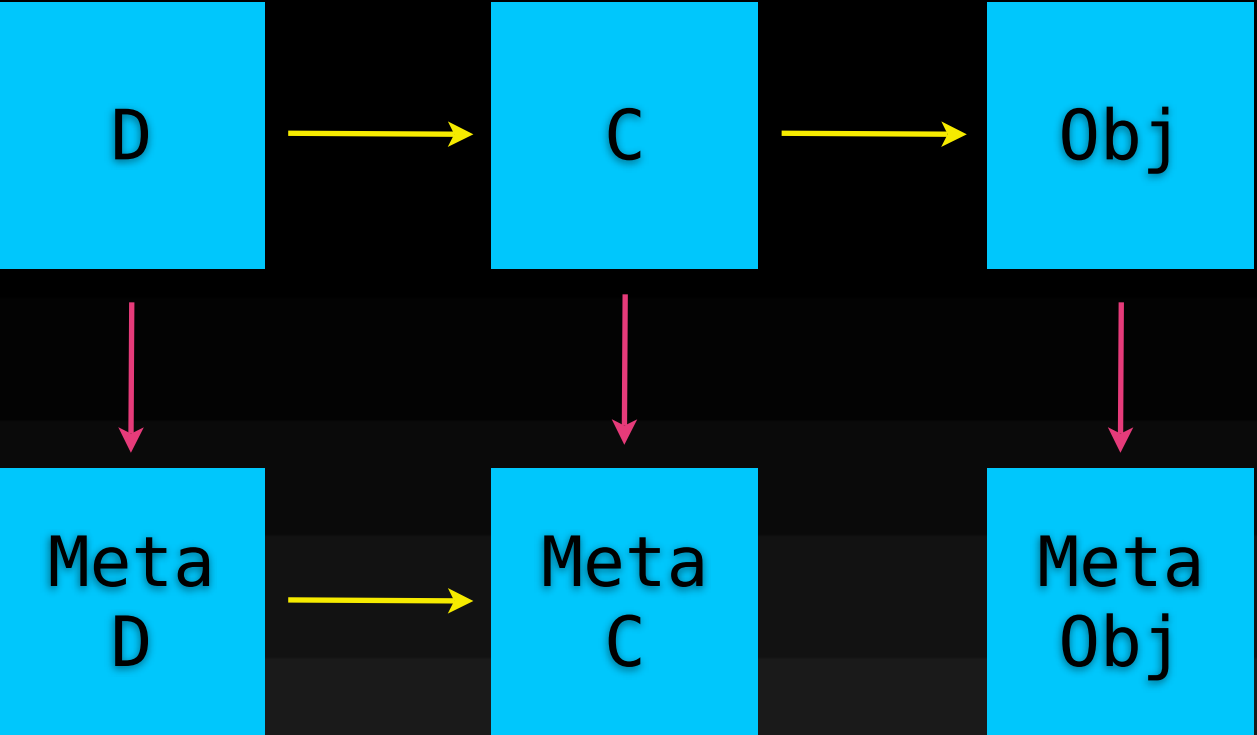
Dispatch

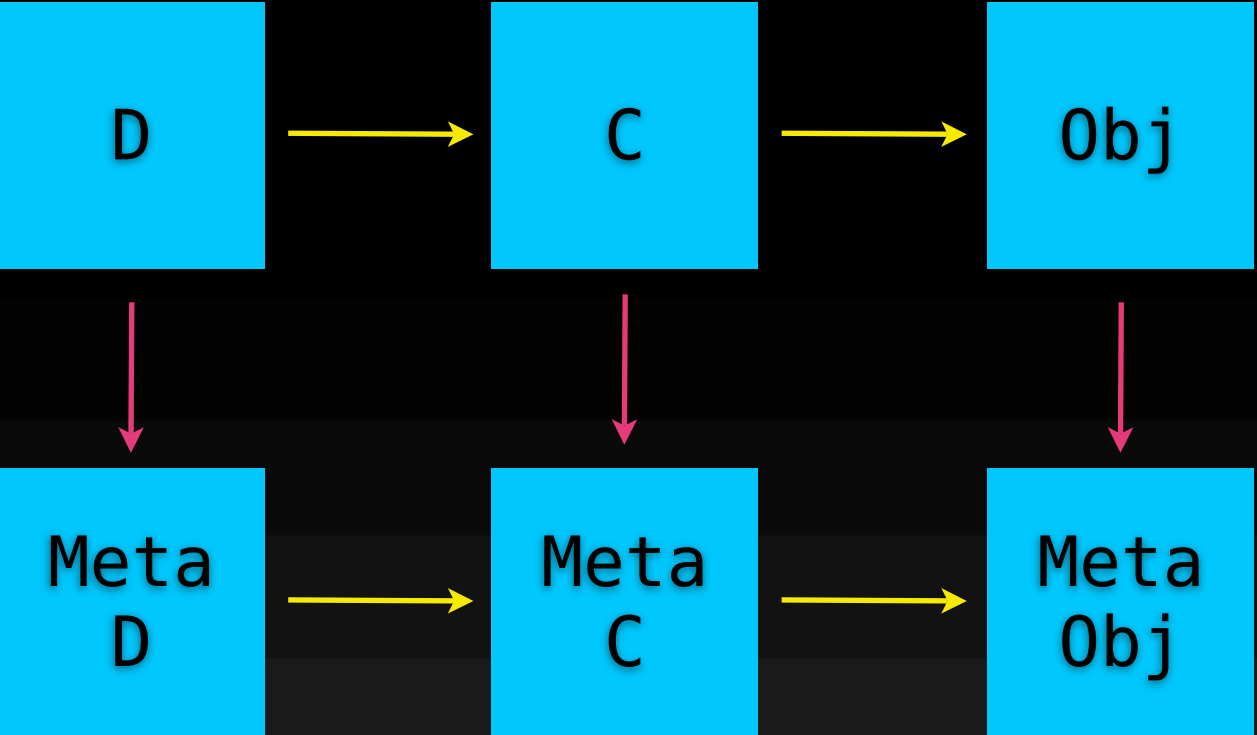


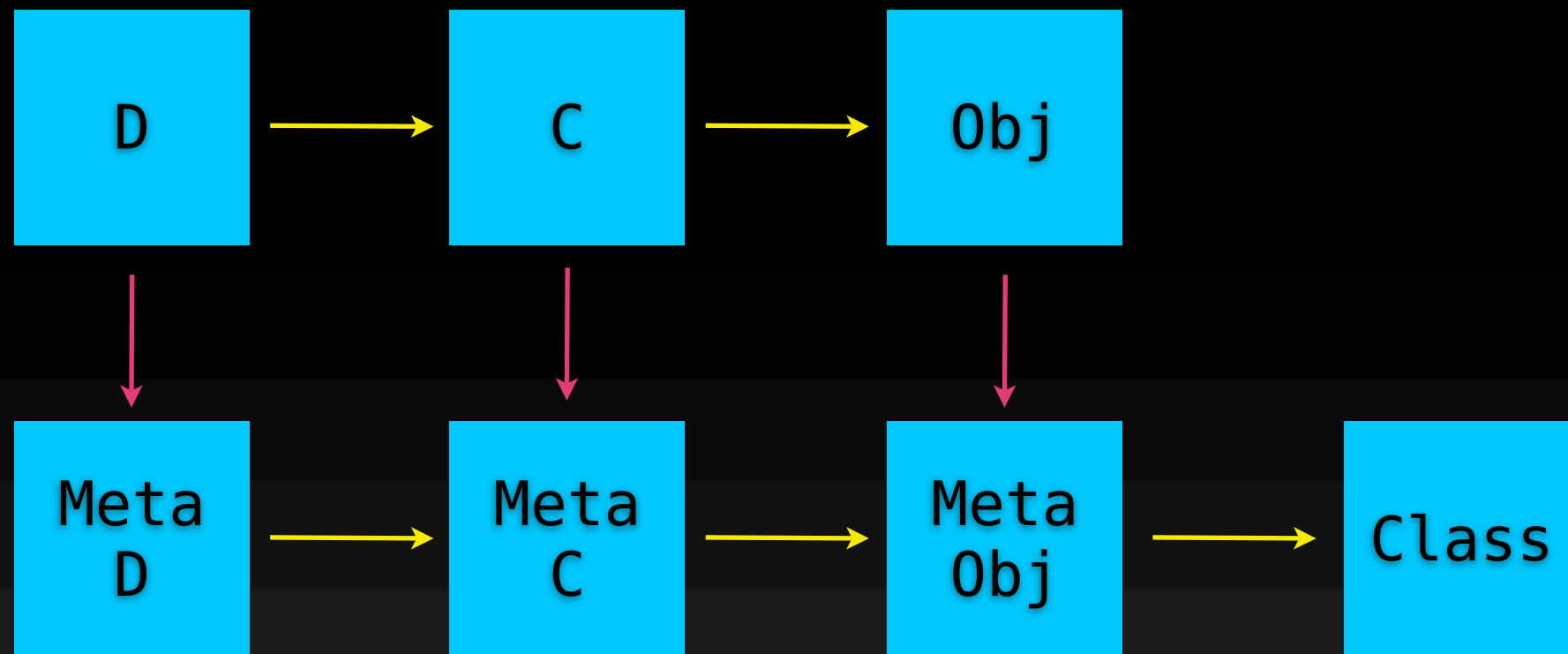


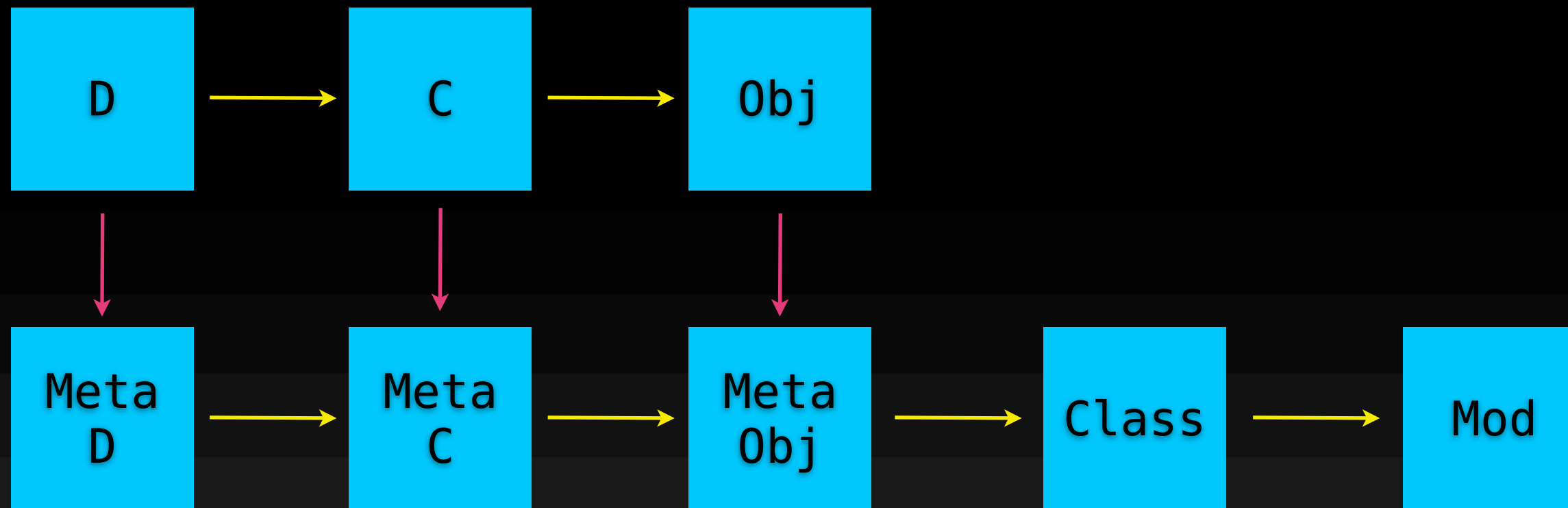


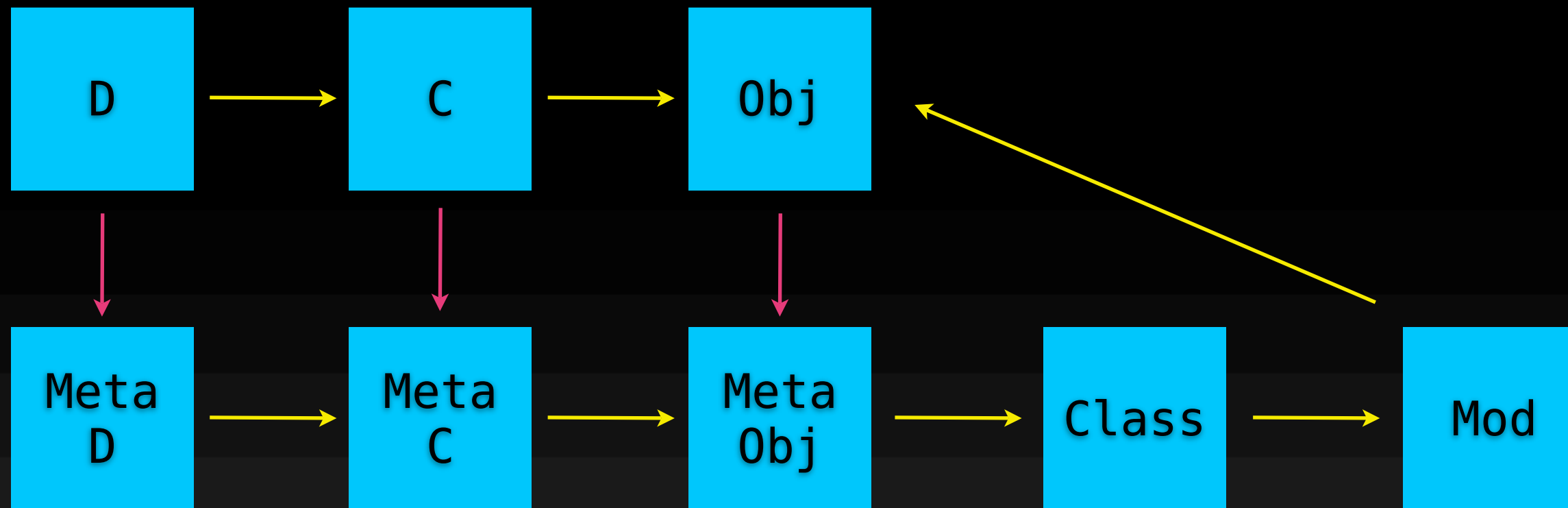


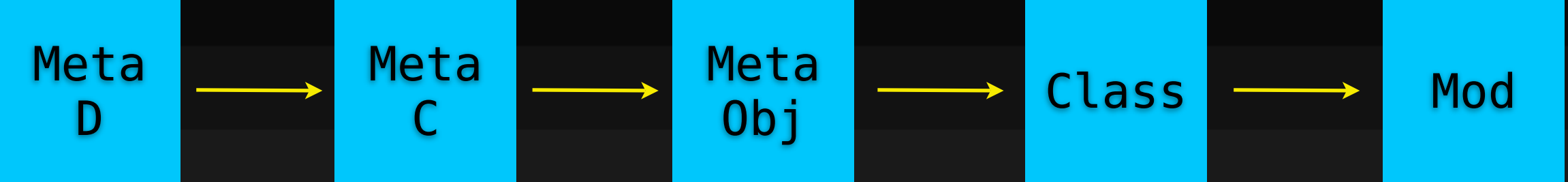


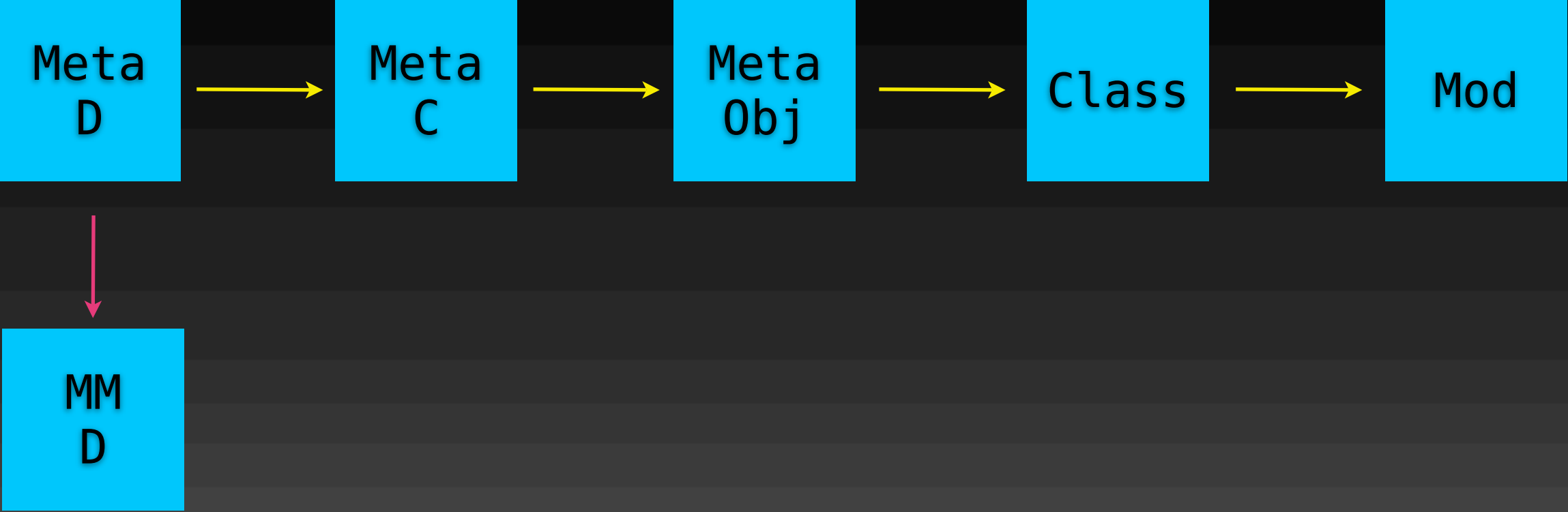


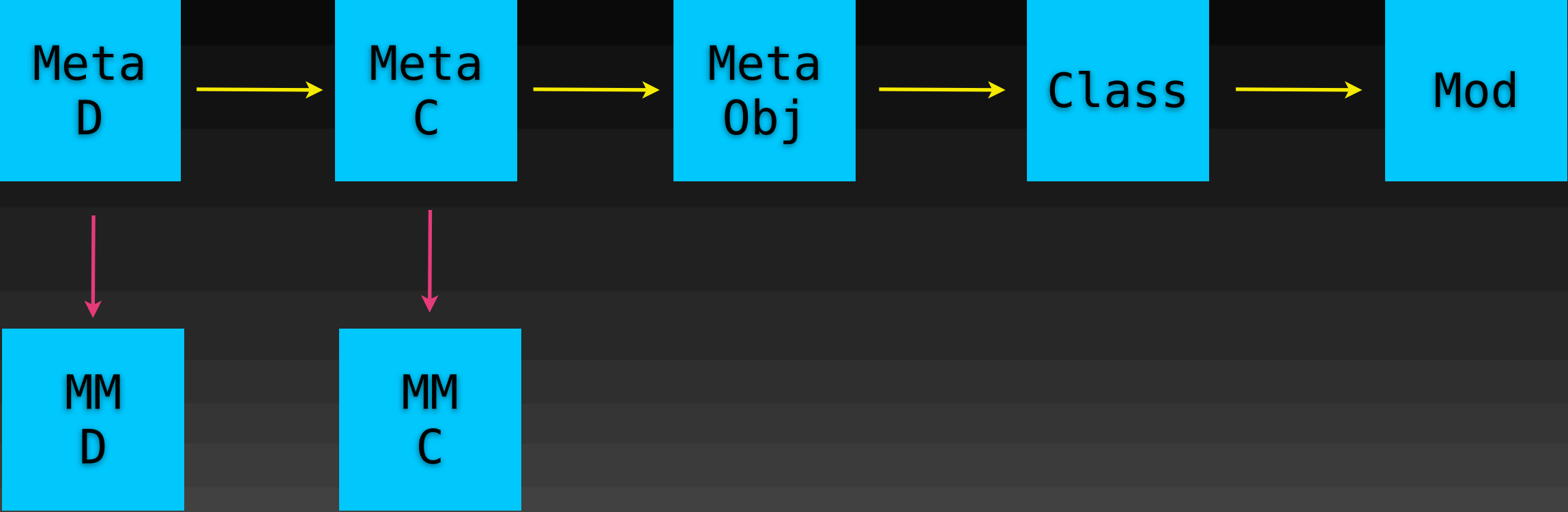


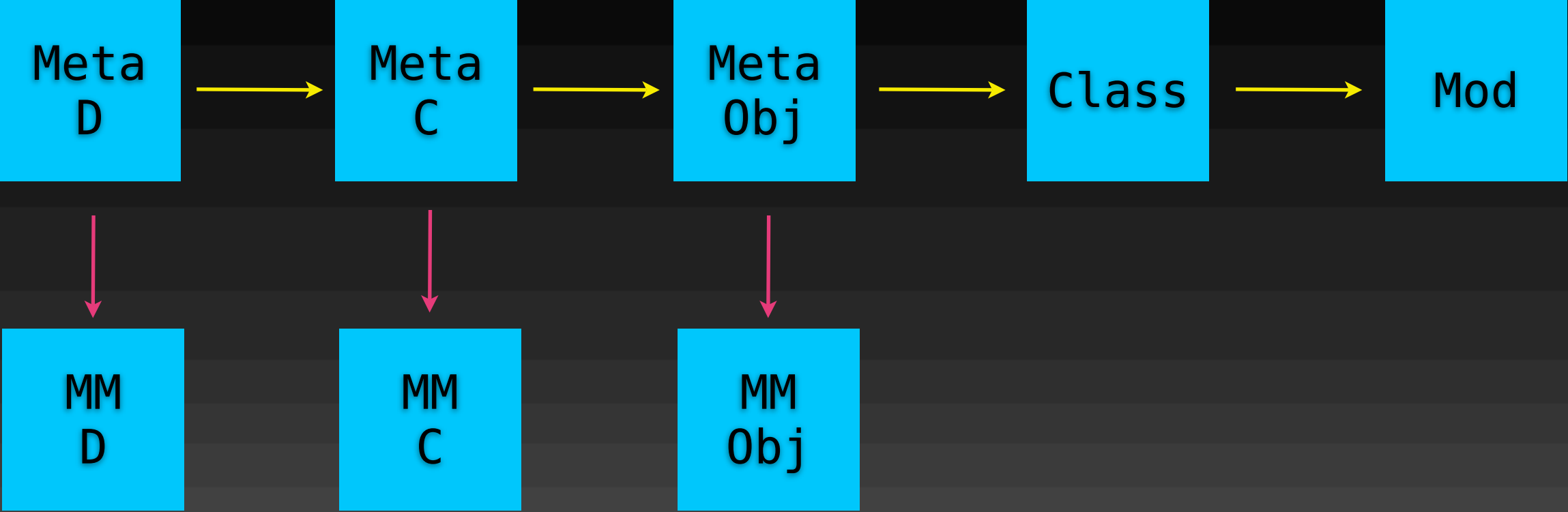


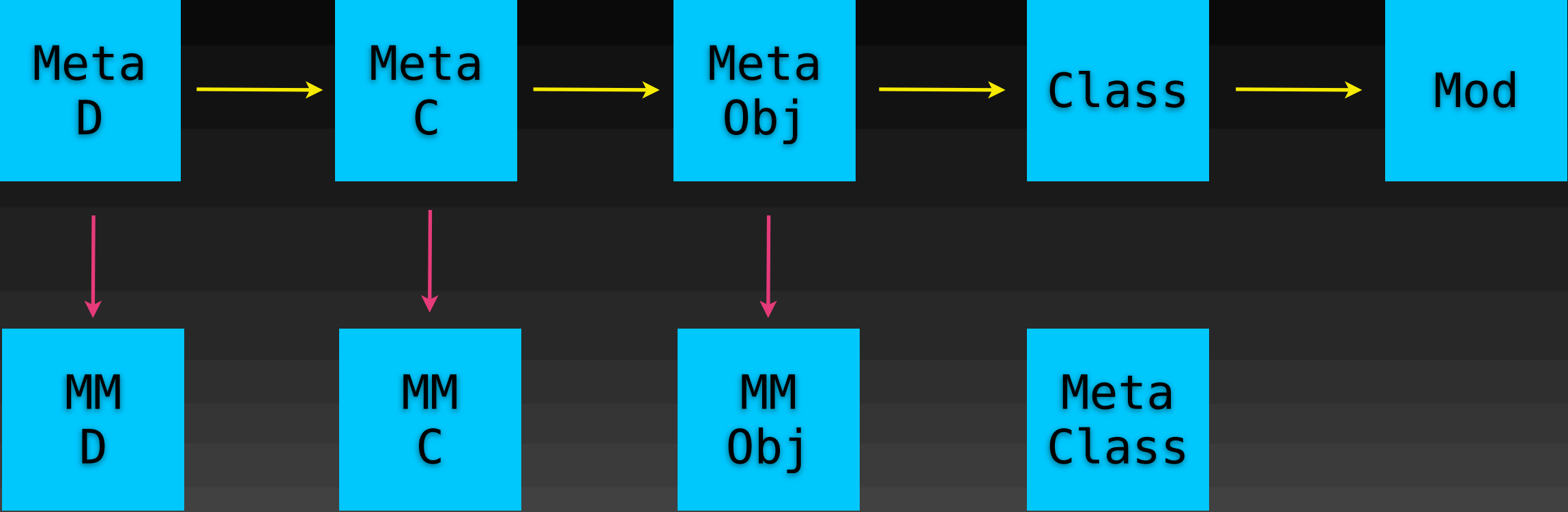


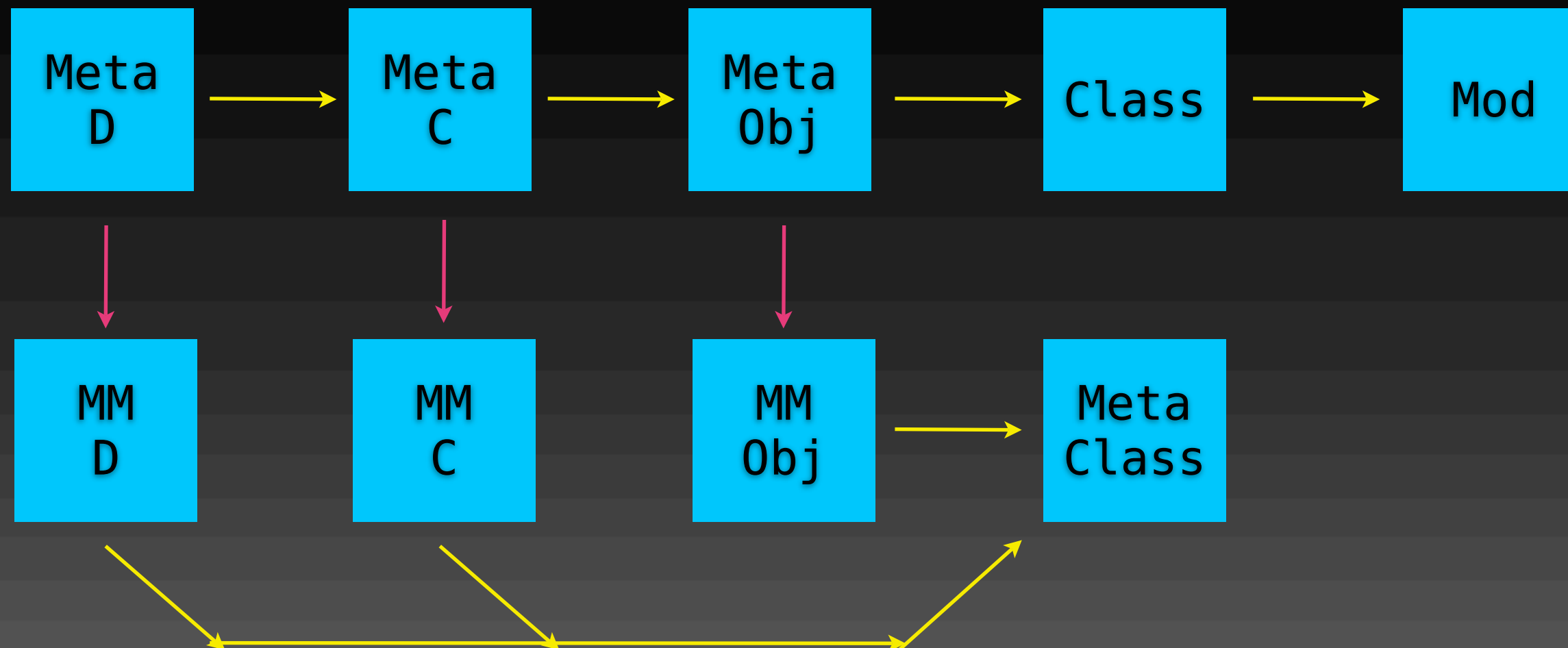


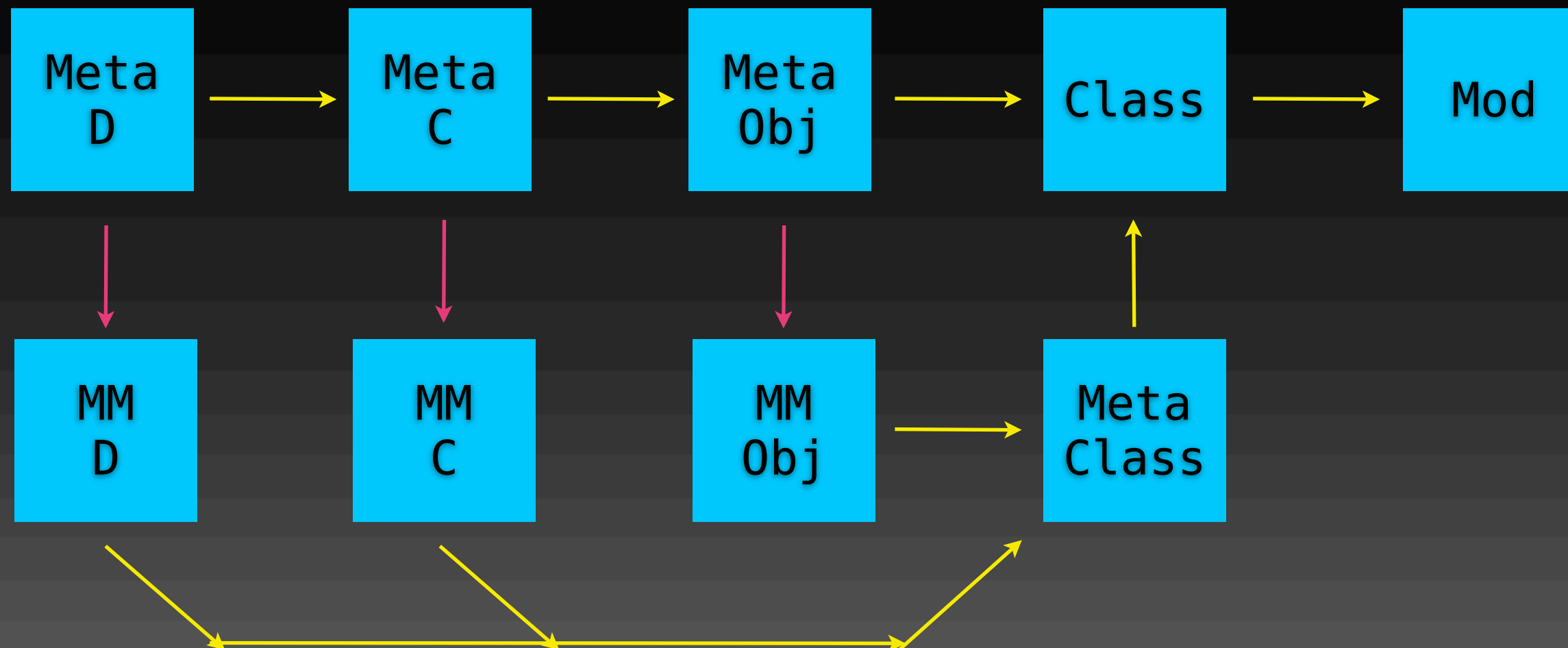


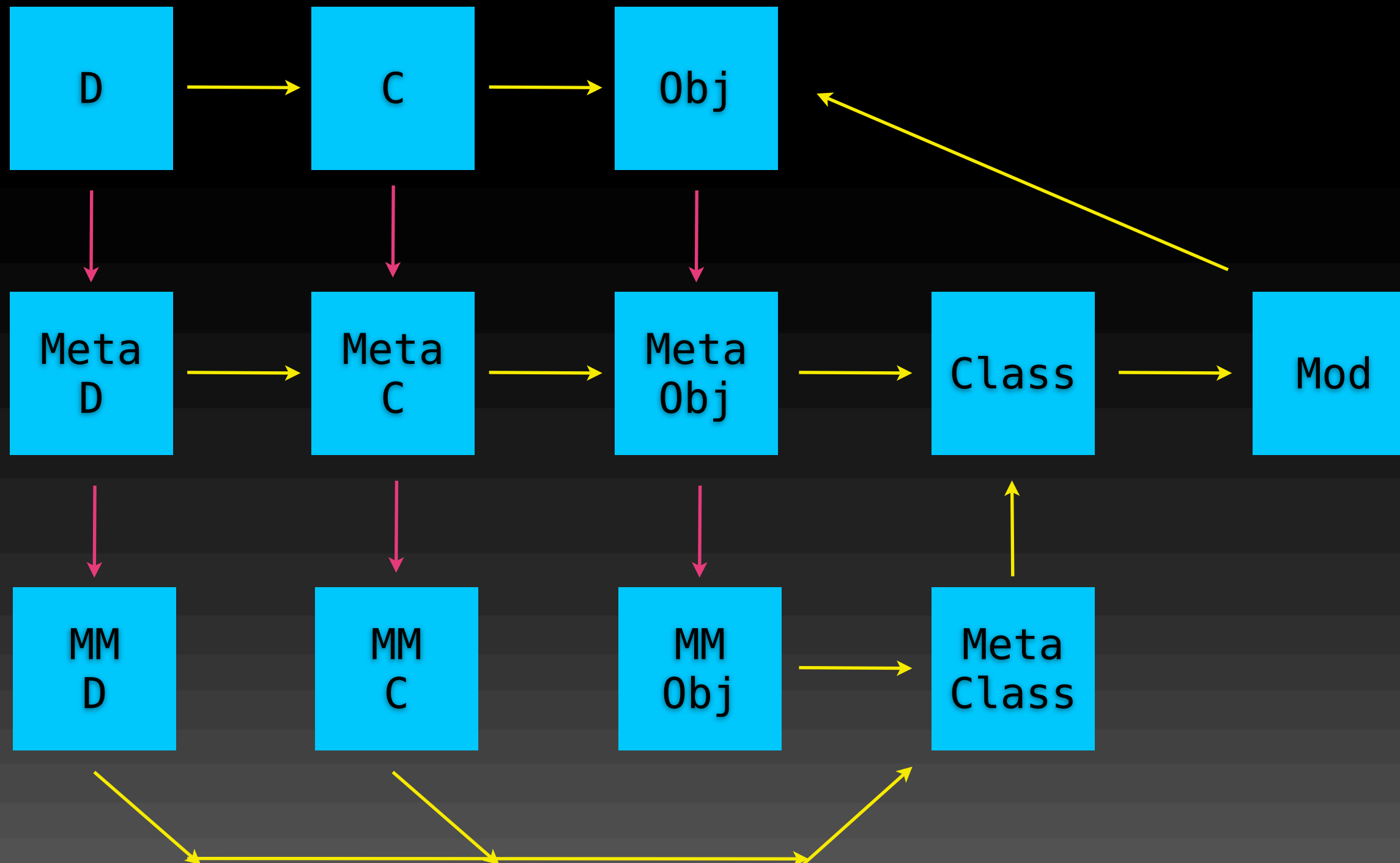


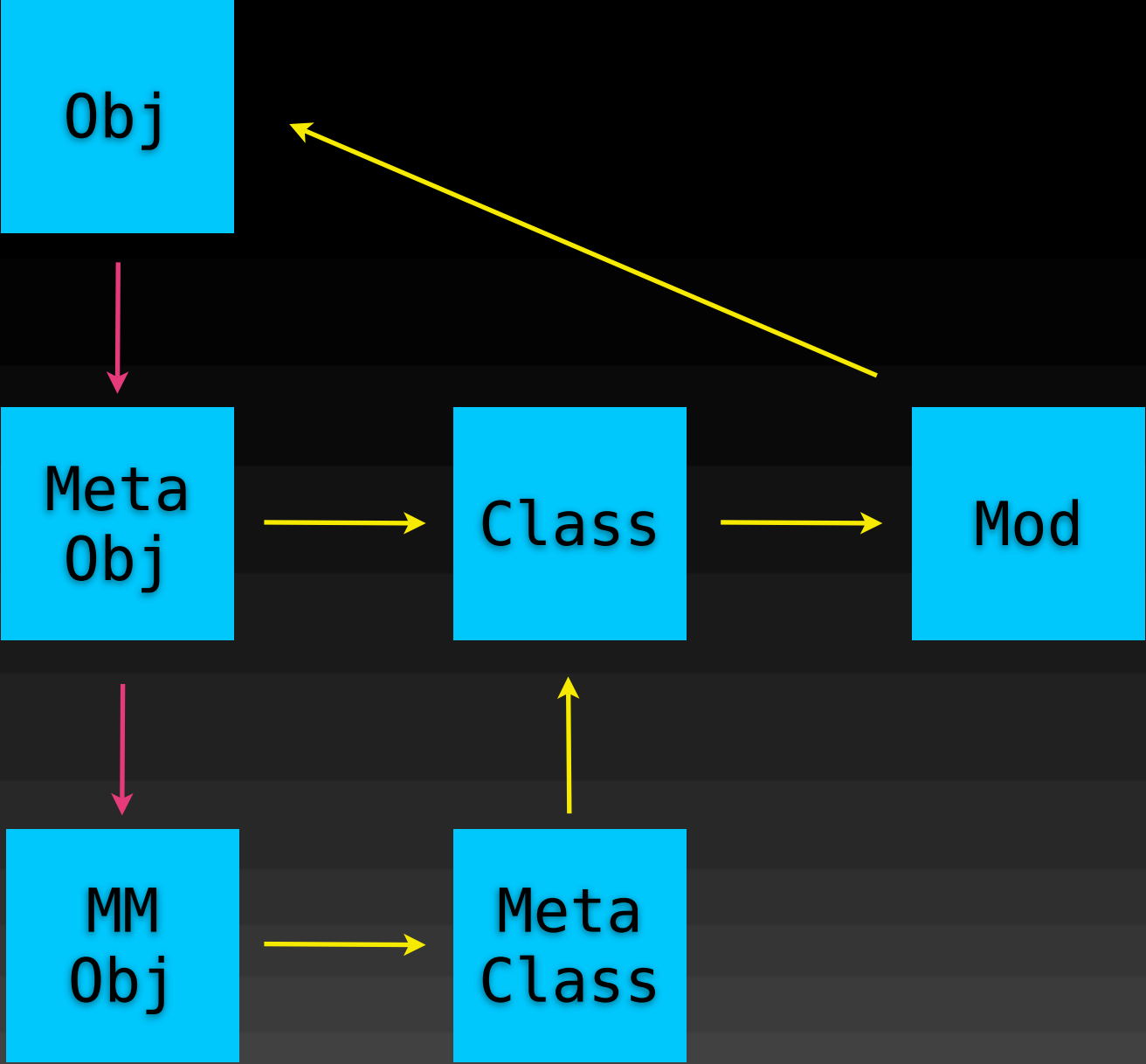


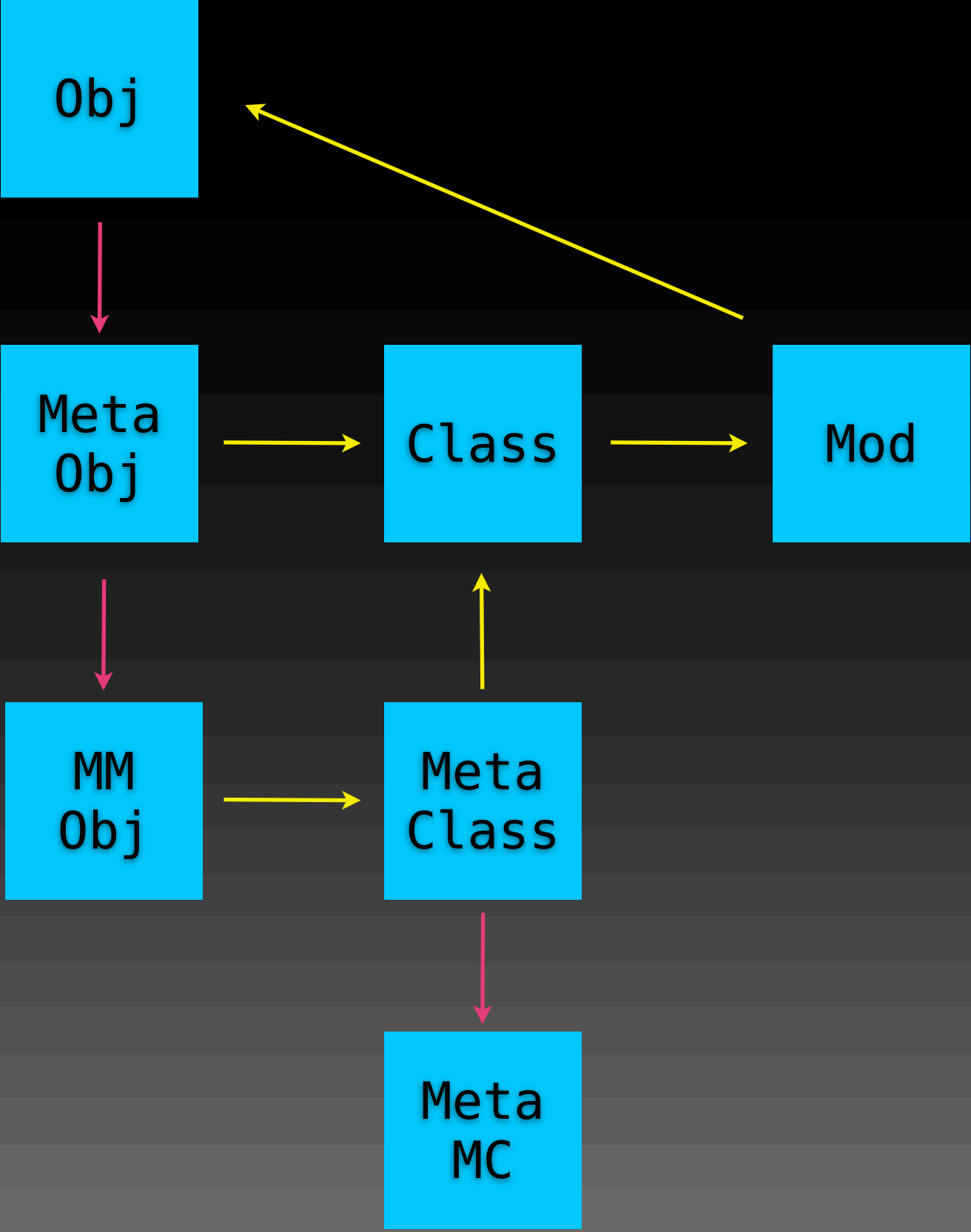


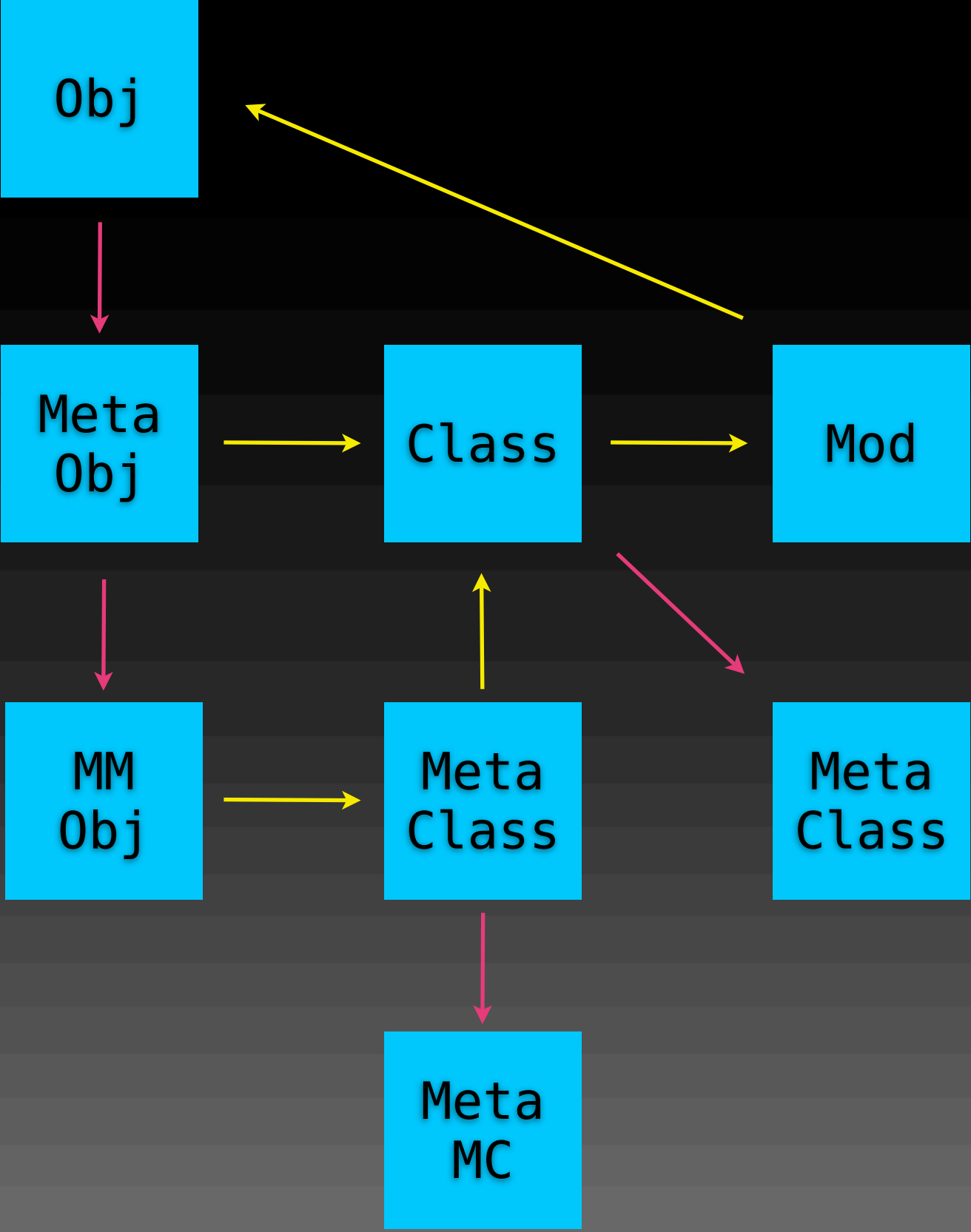


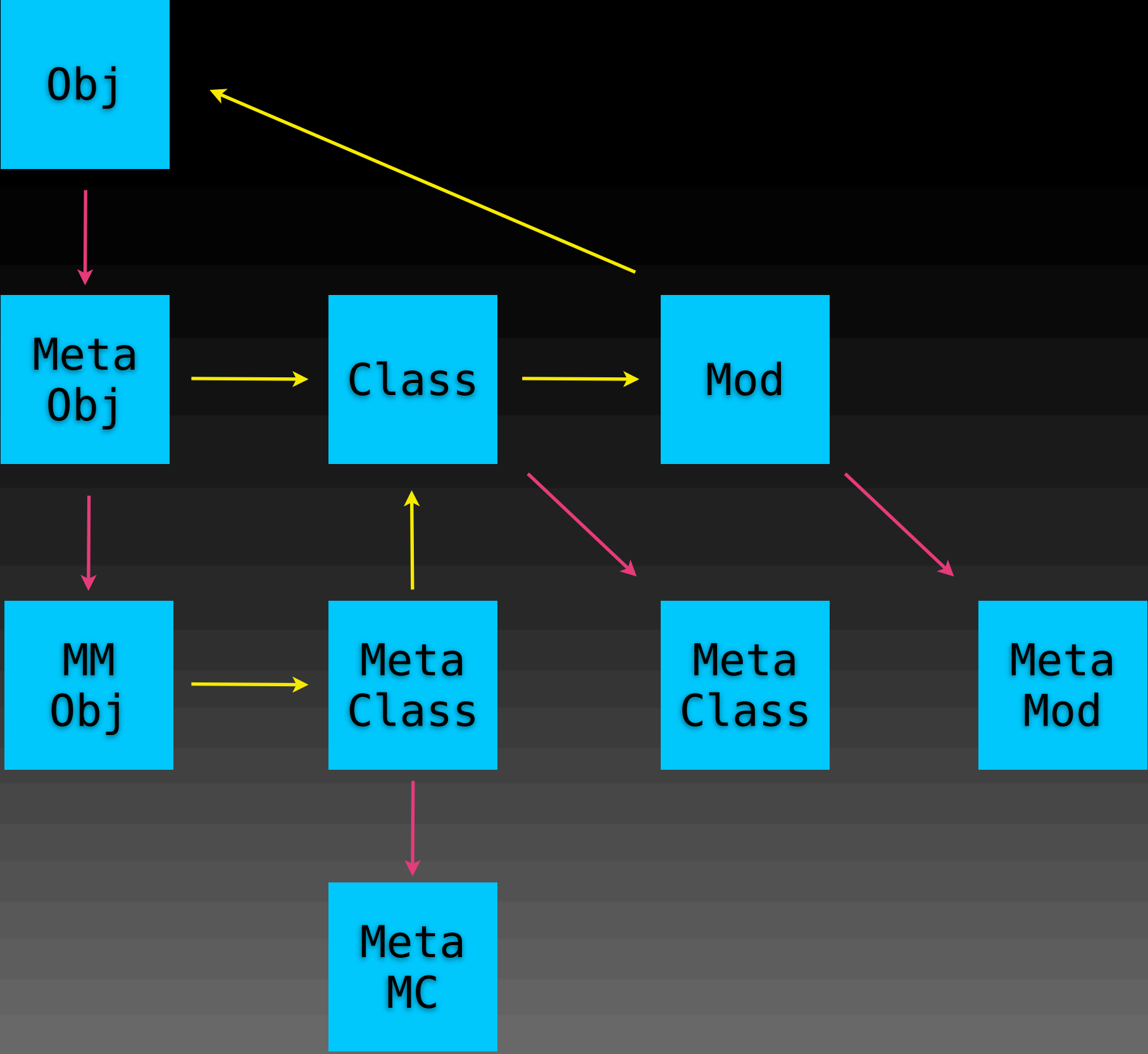


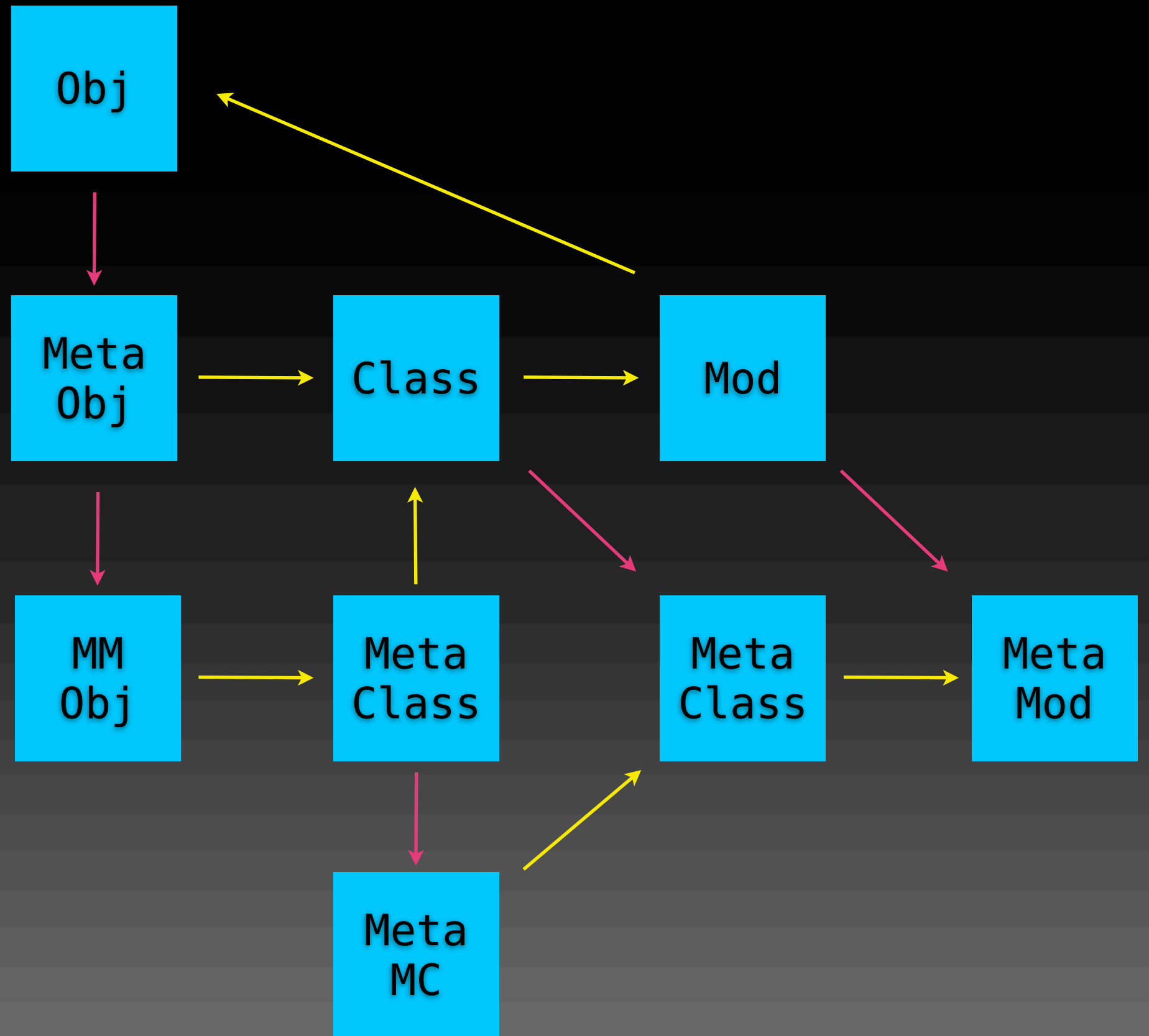


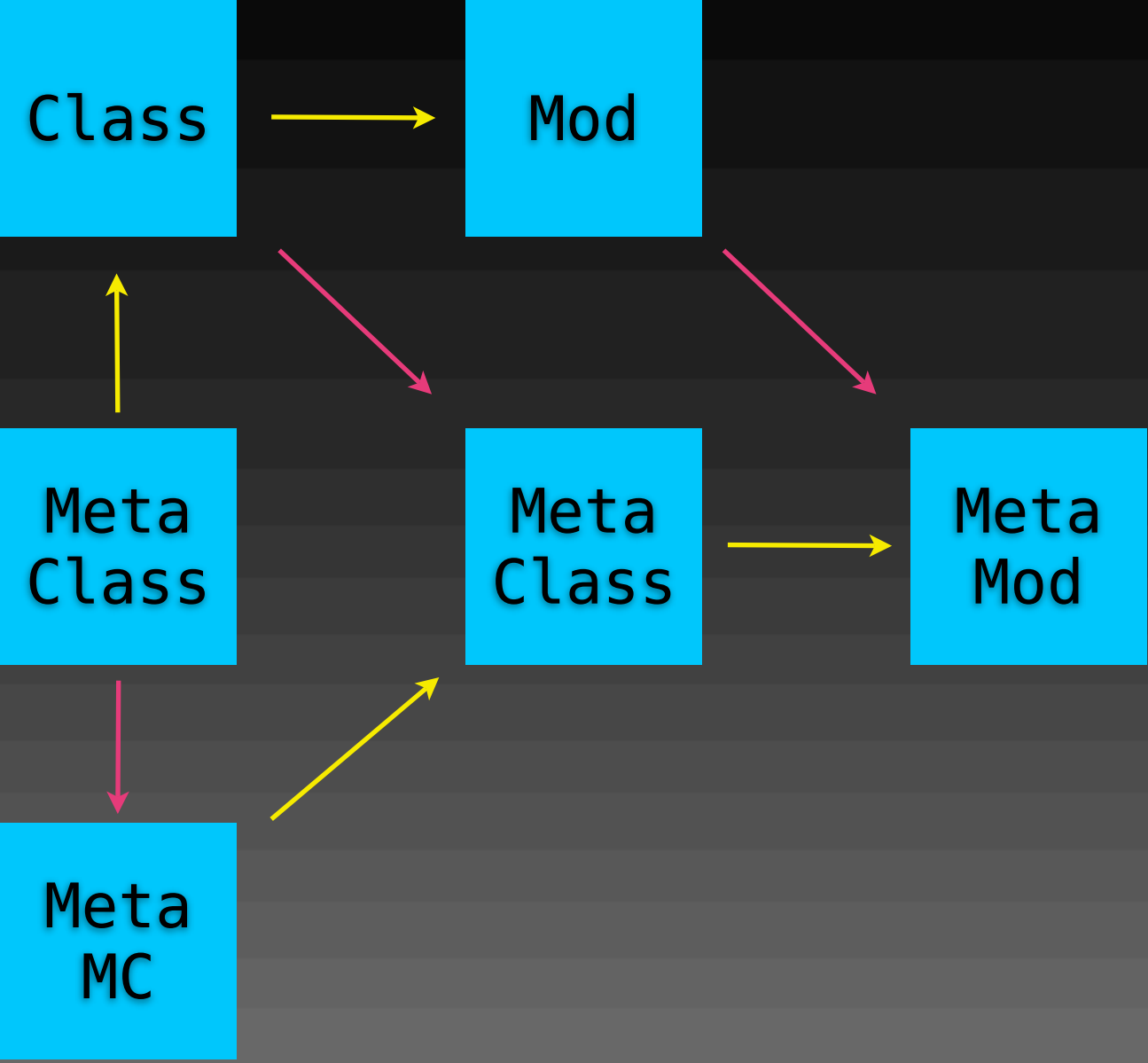












Meta
Class

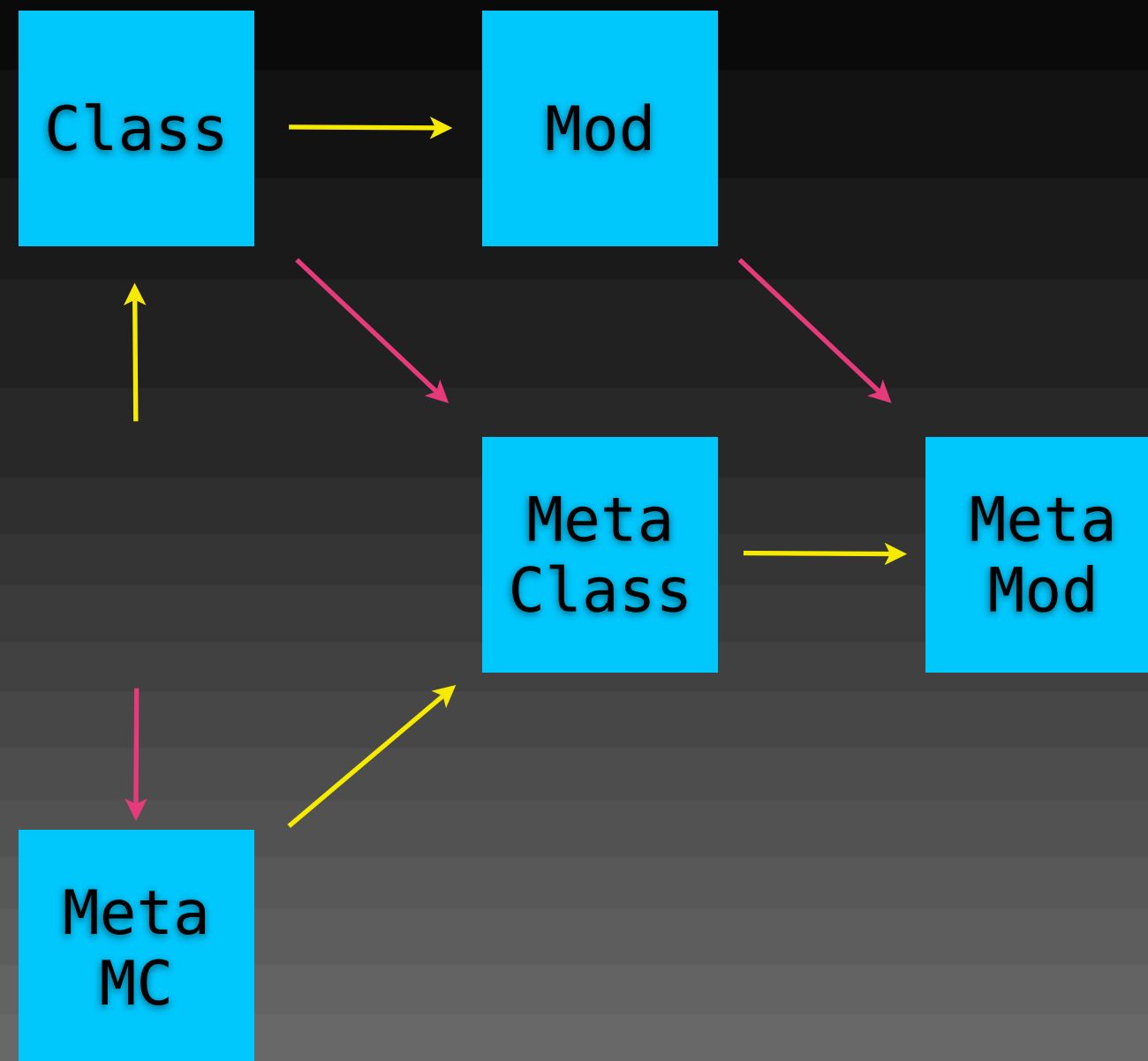
Class

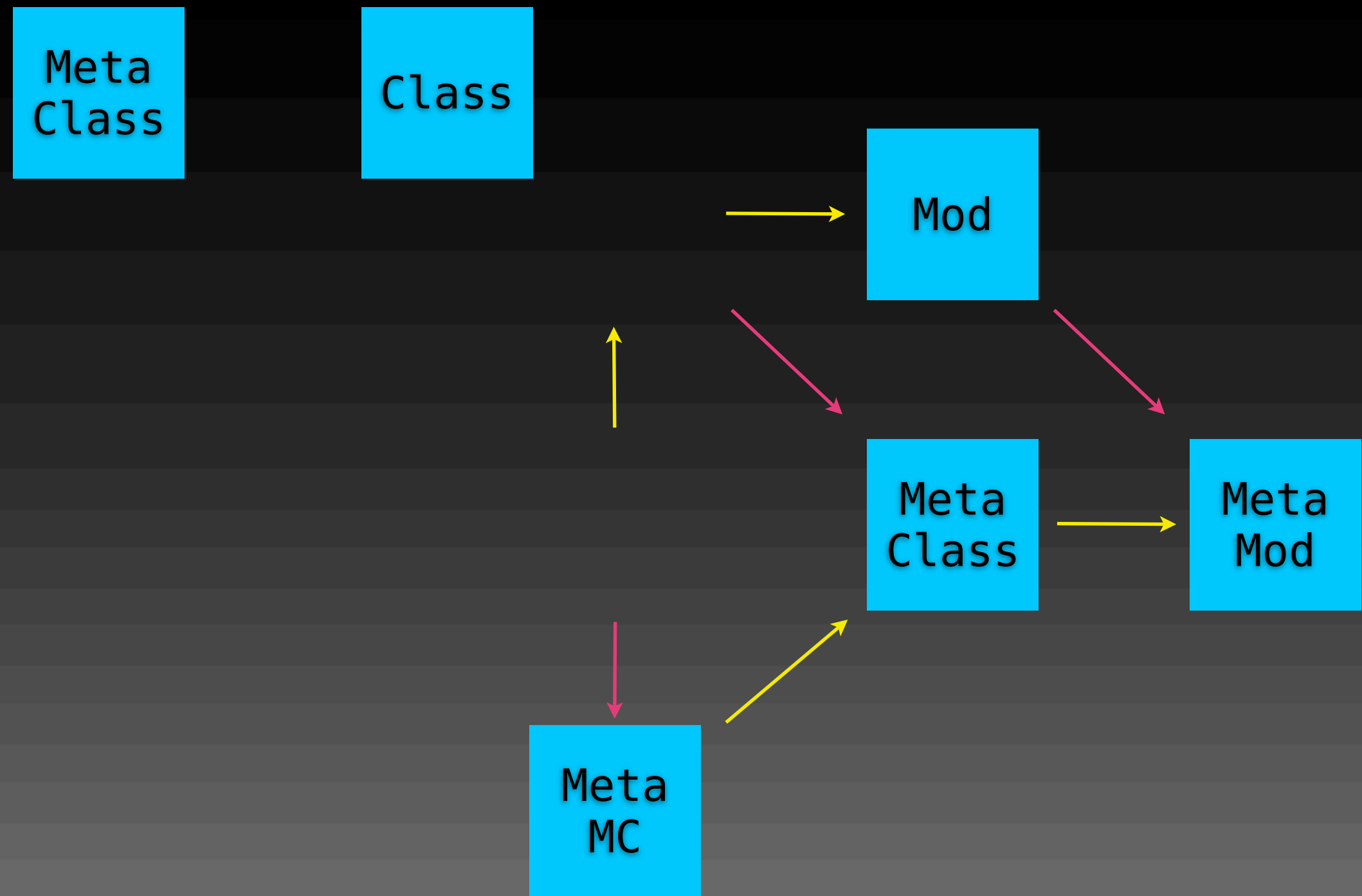
Mod

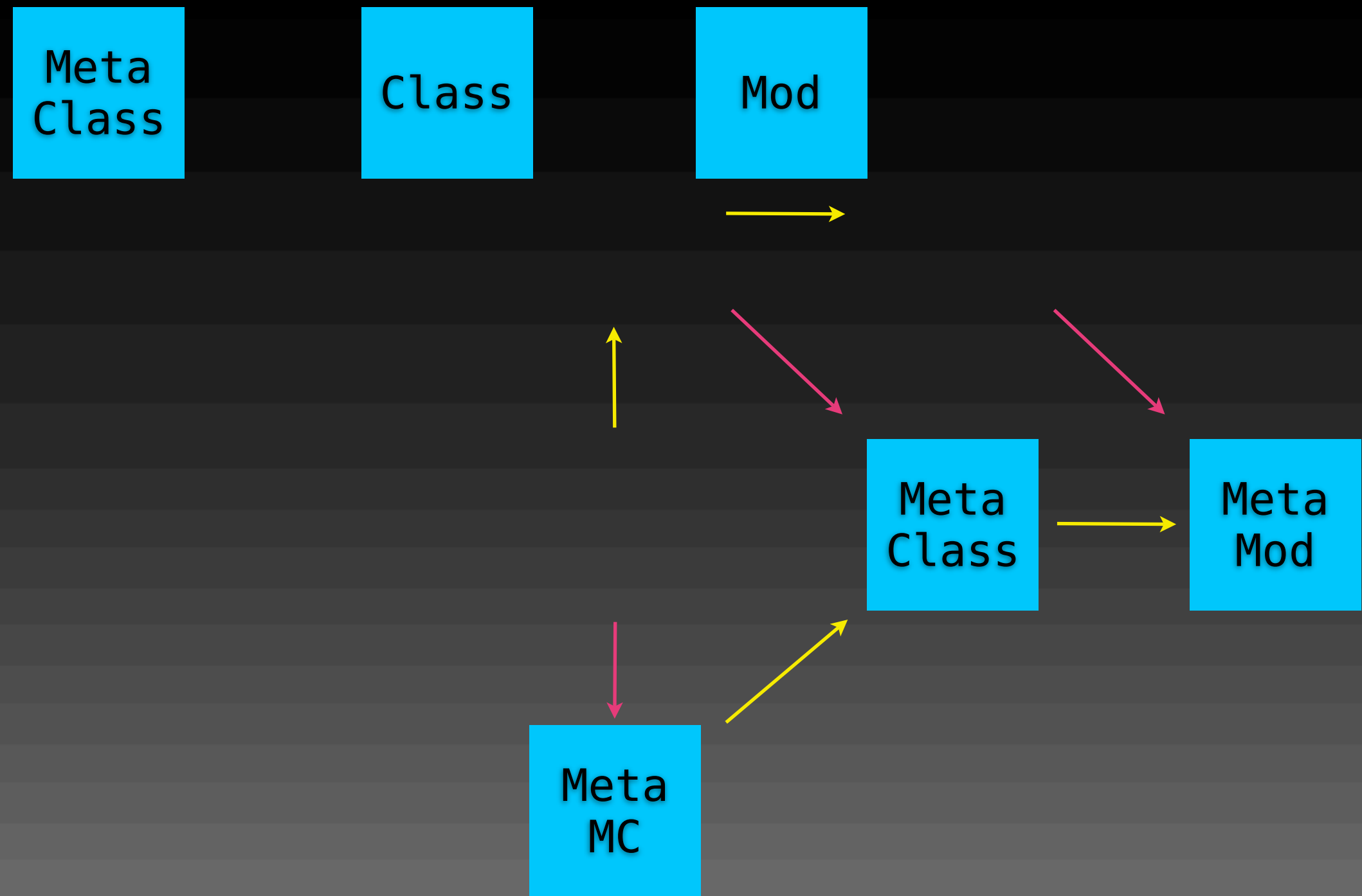
Meta
Class

Meta
Mod

Meta
MC







Meta
Class

Class

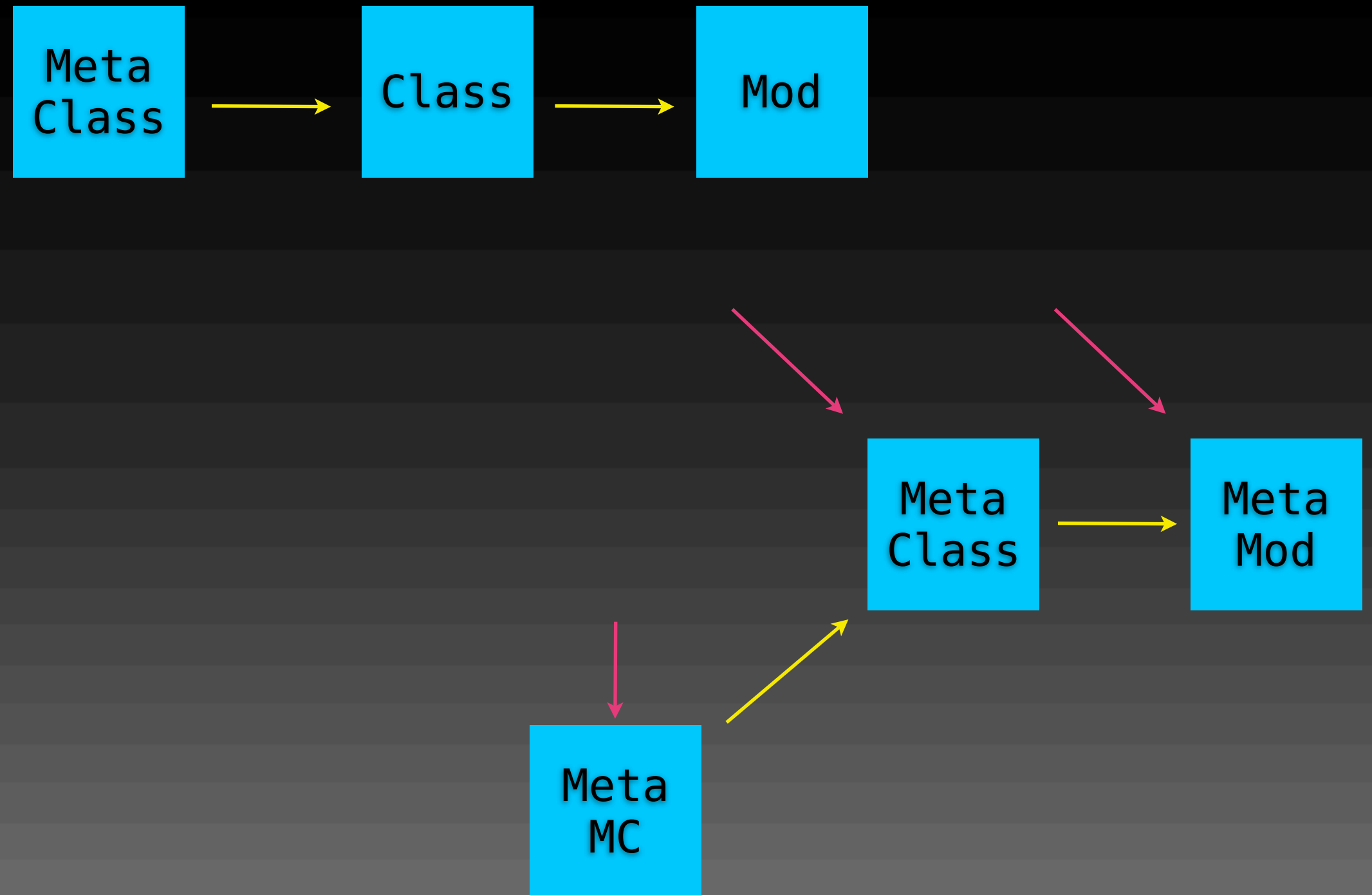
Mod

Meta
Class

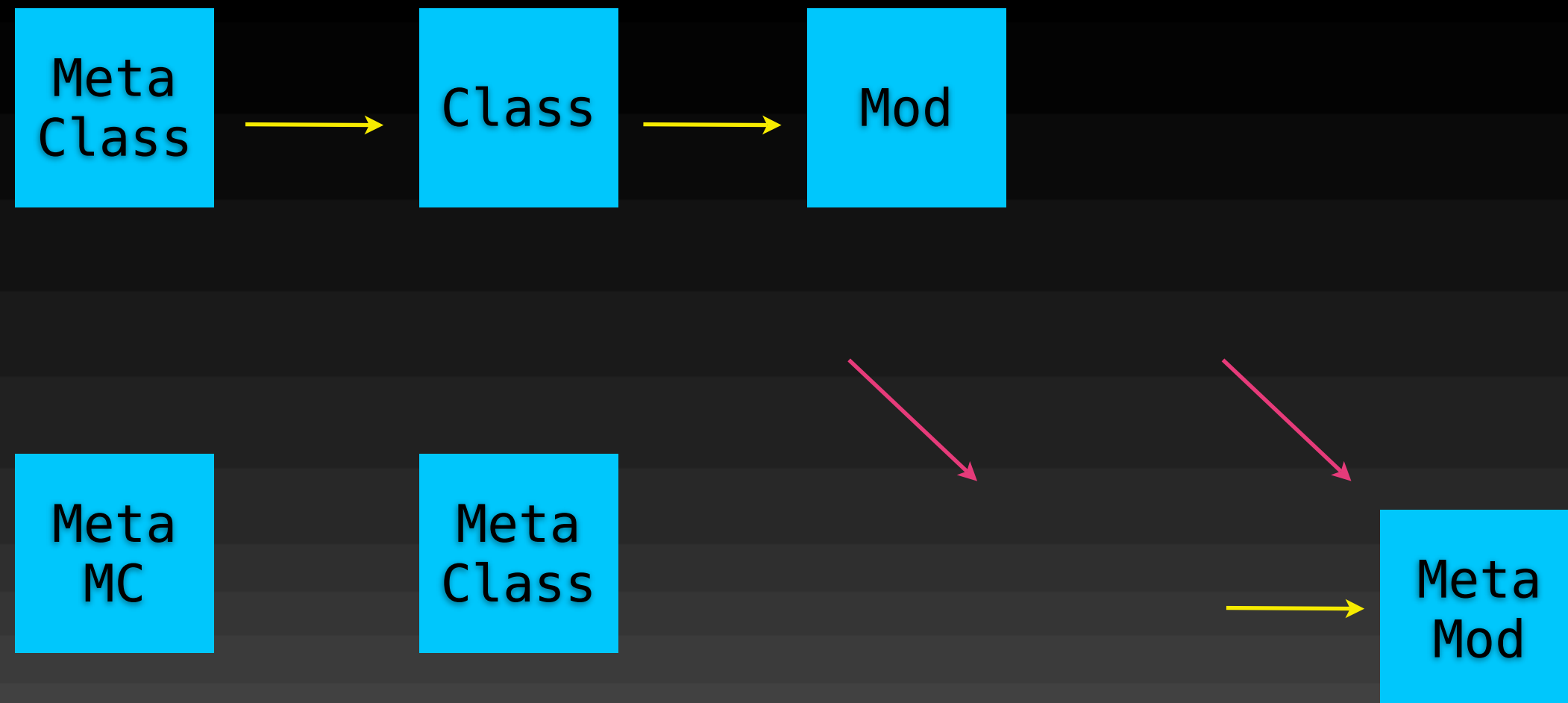
Meta
Mod

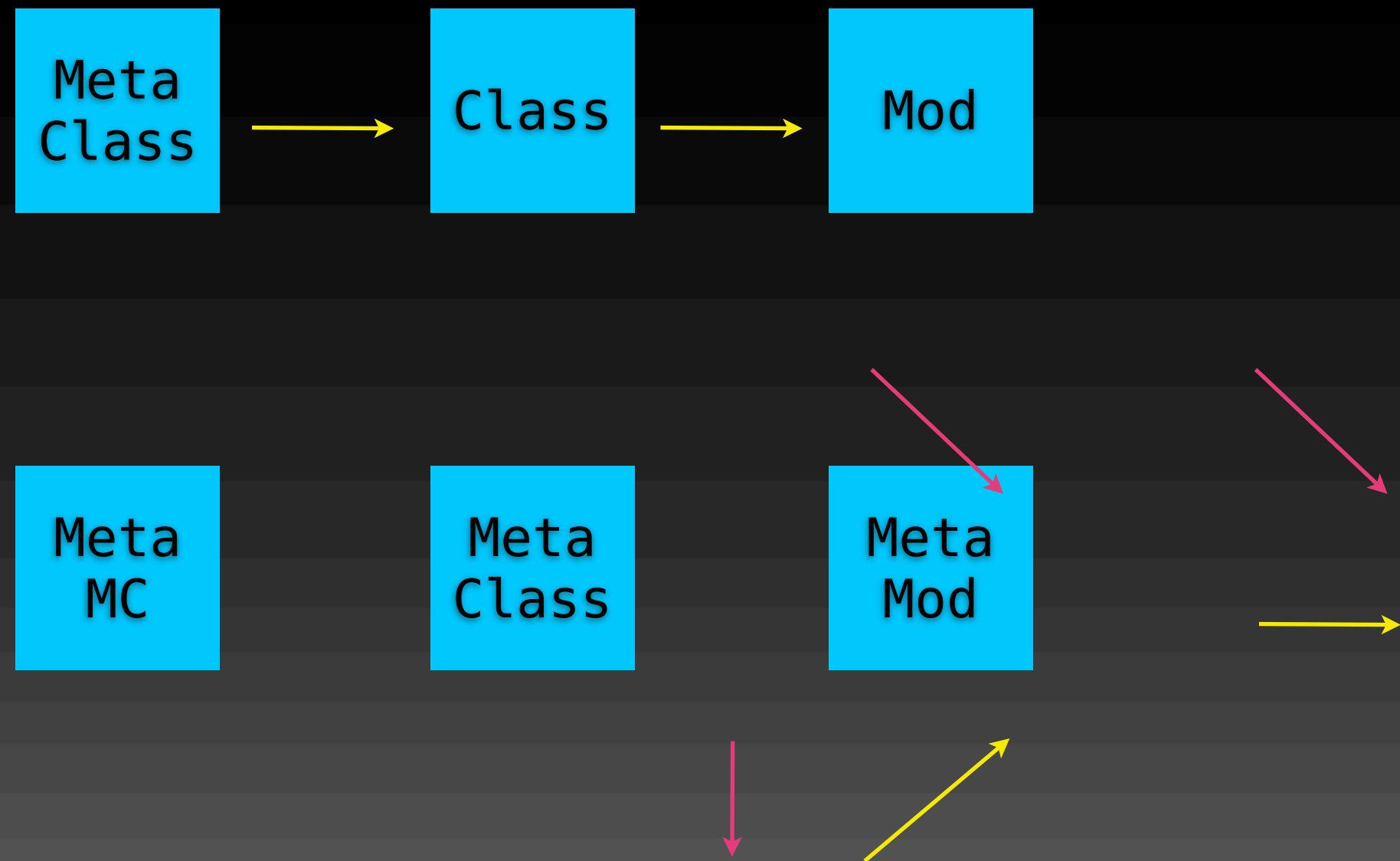
Meta
MC

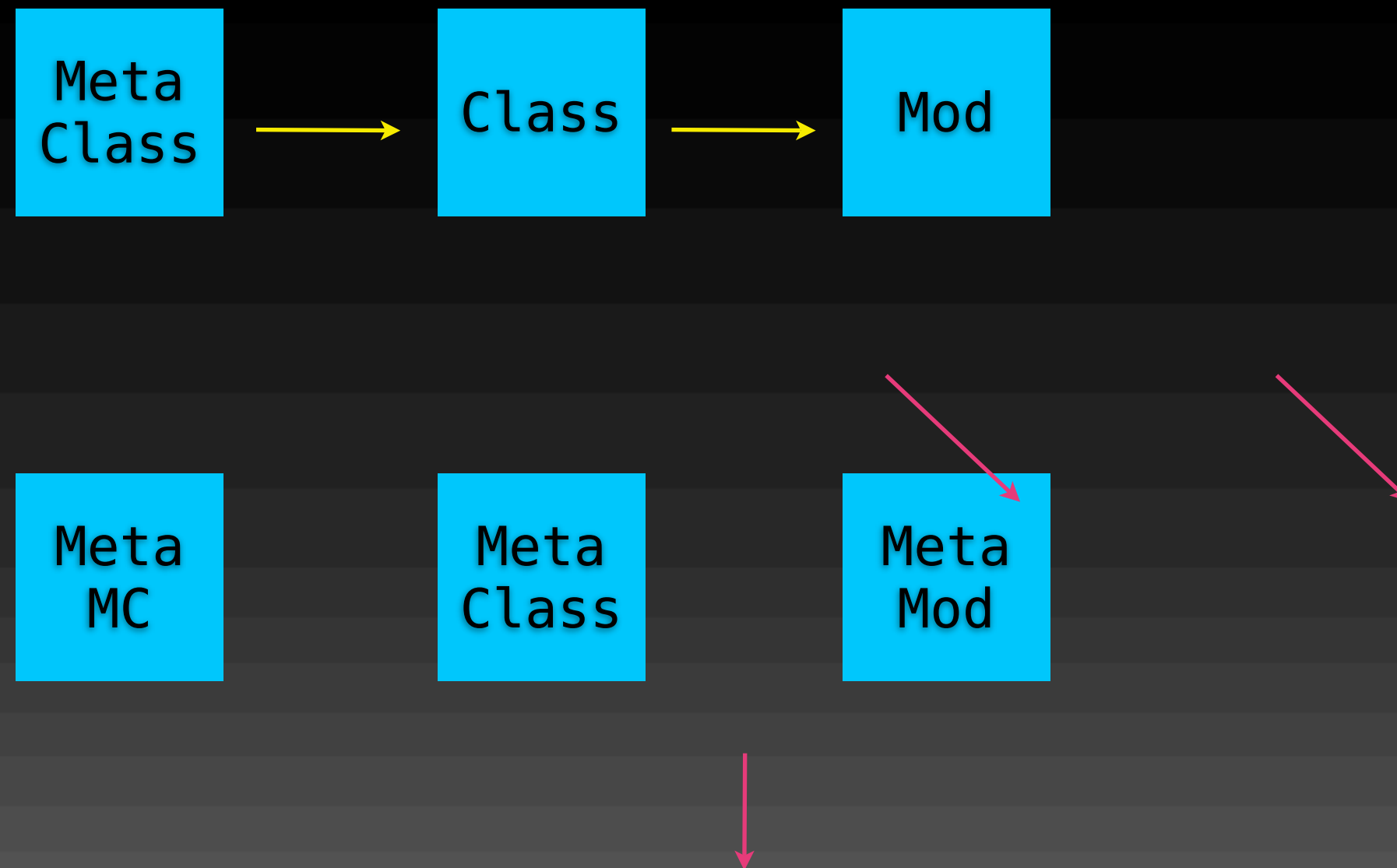


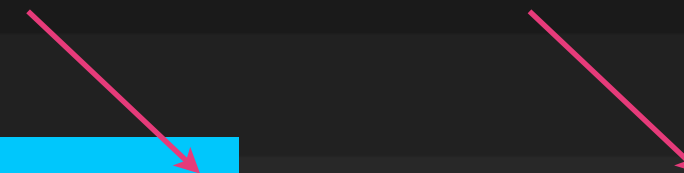
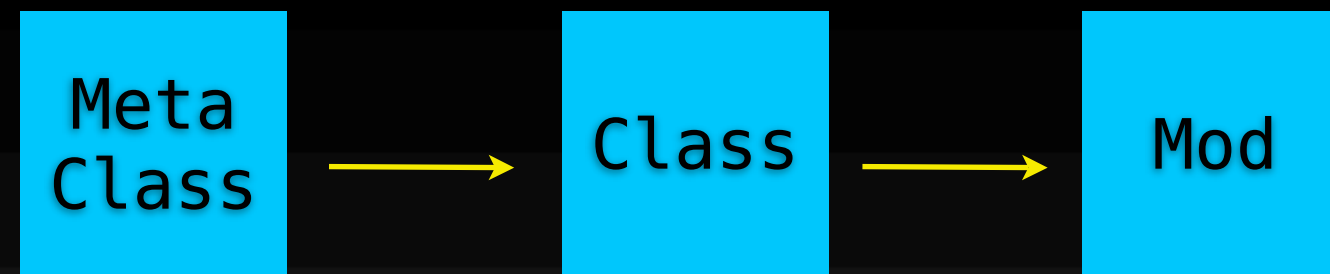




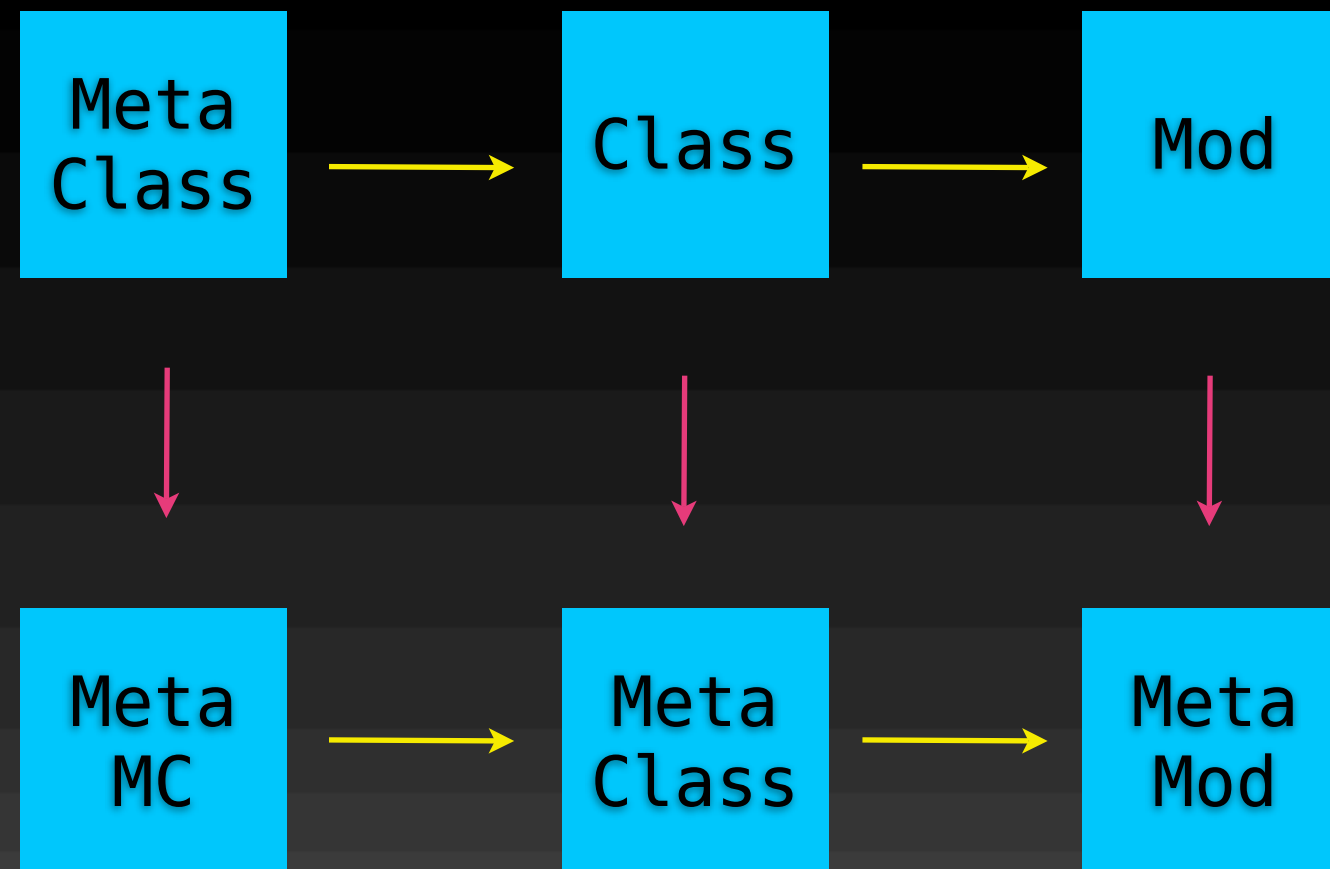


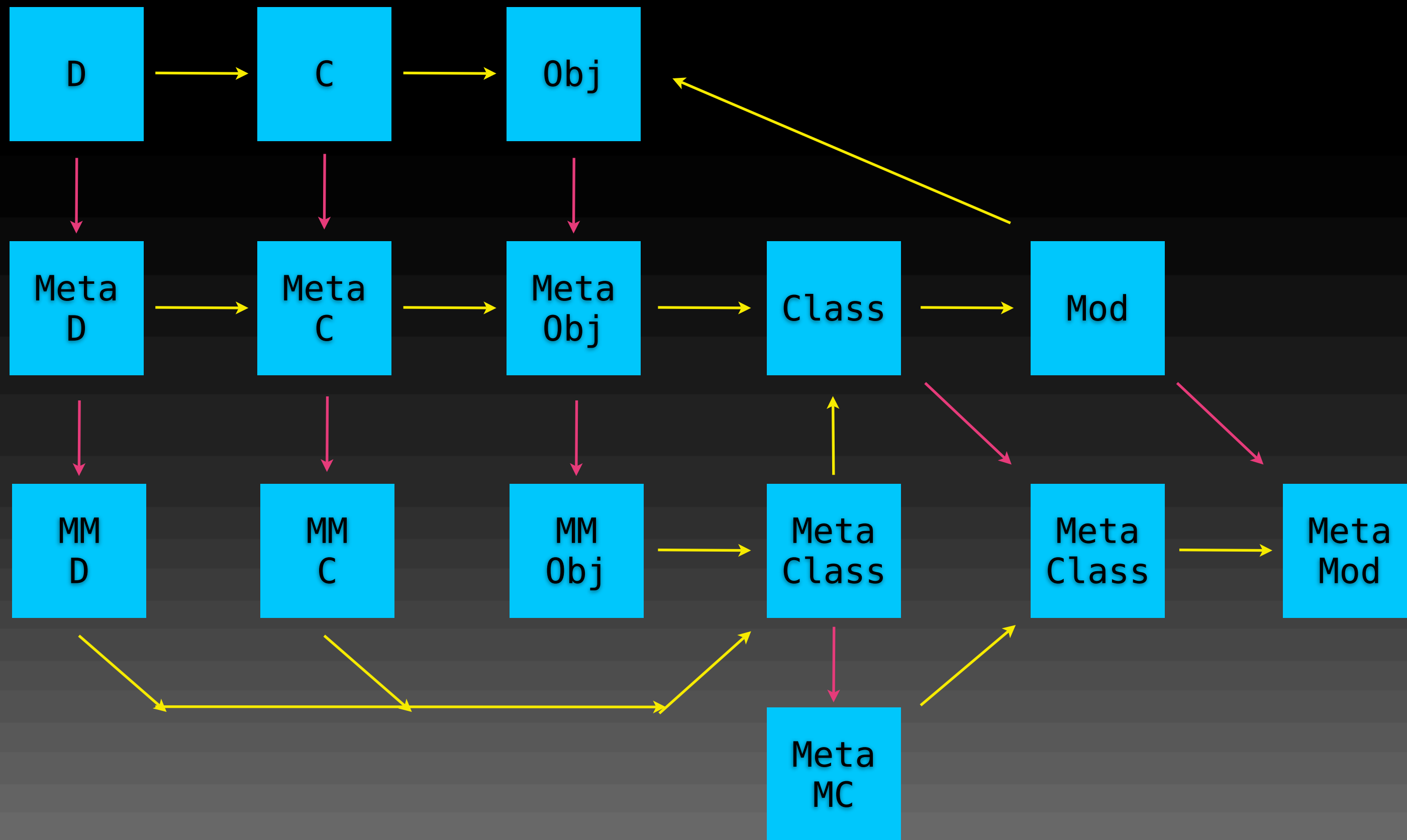


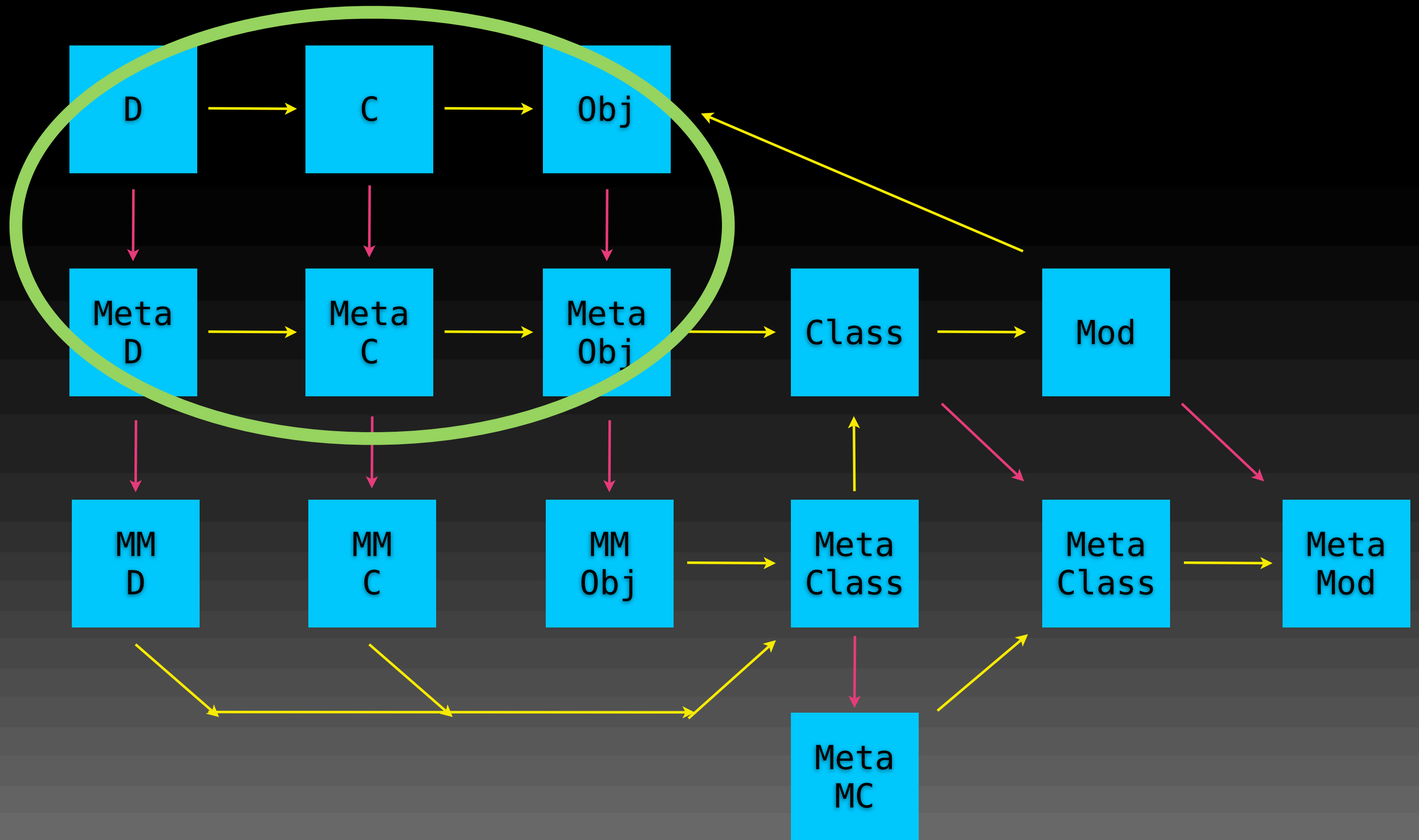


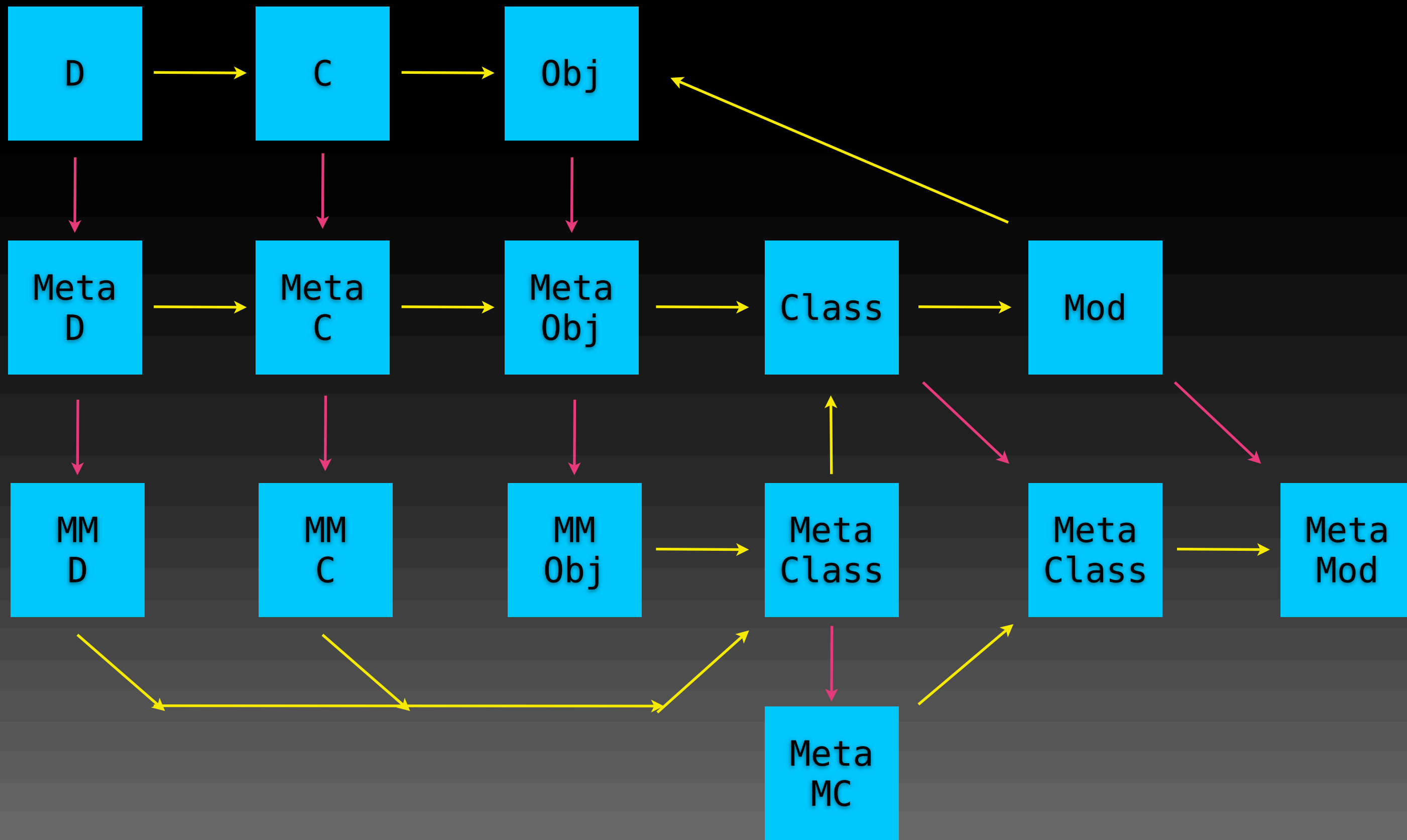


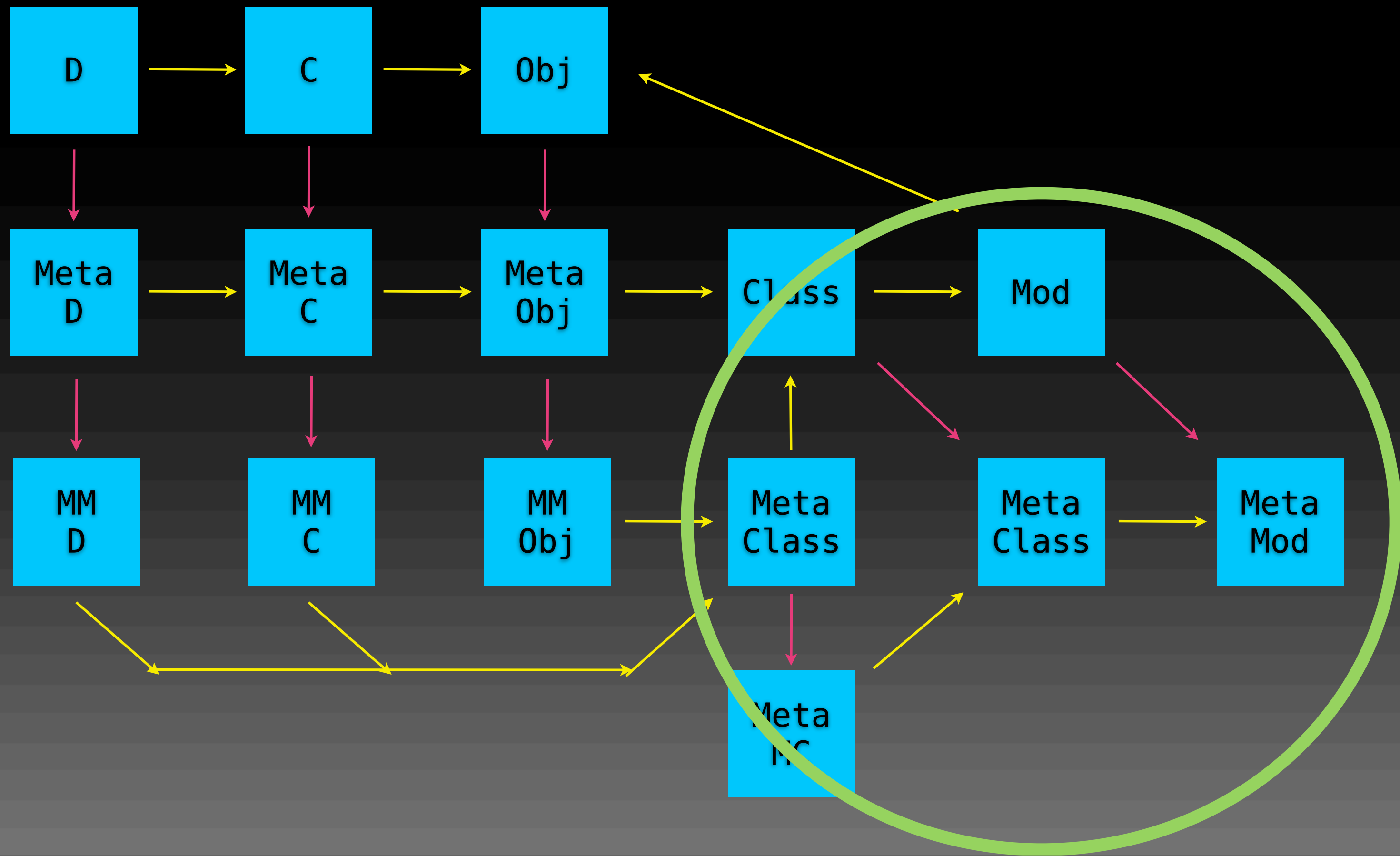


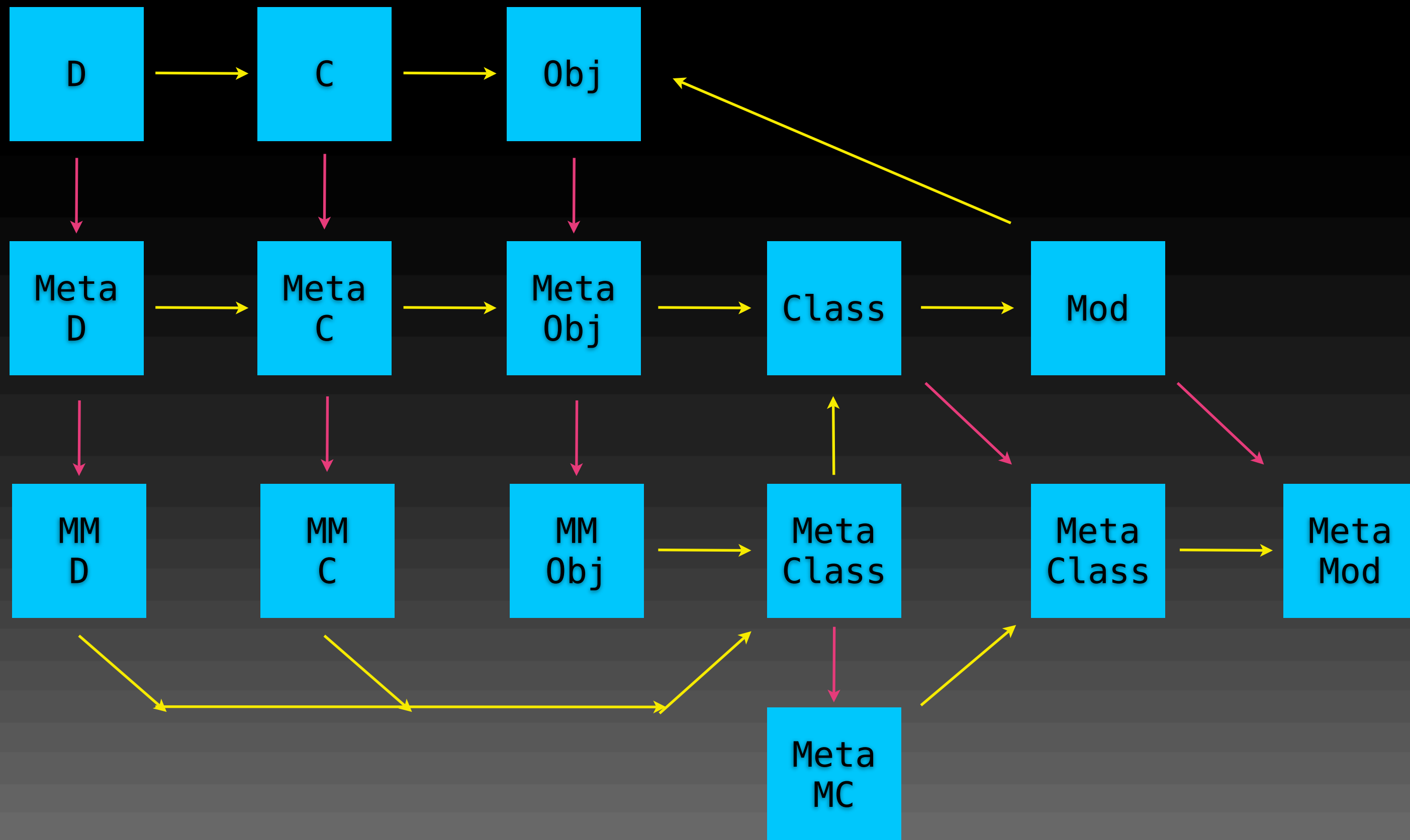


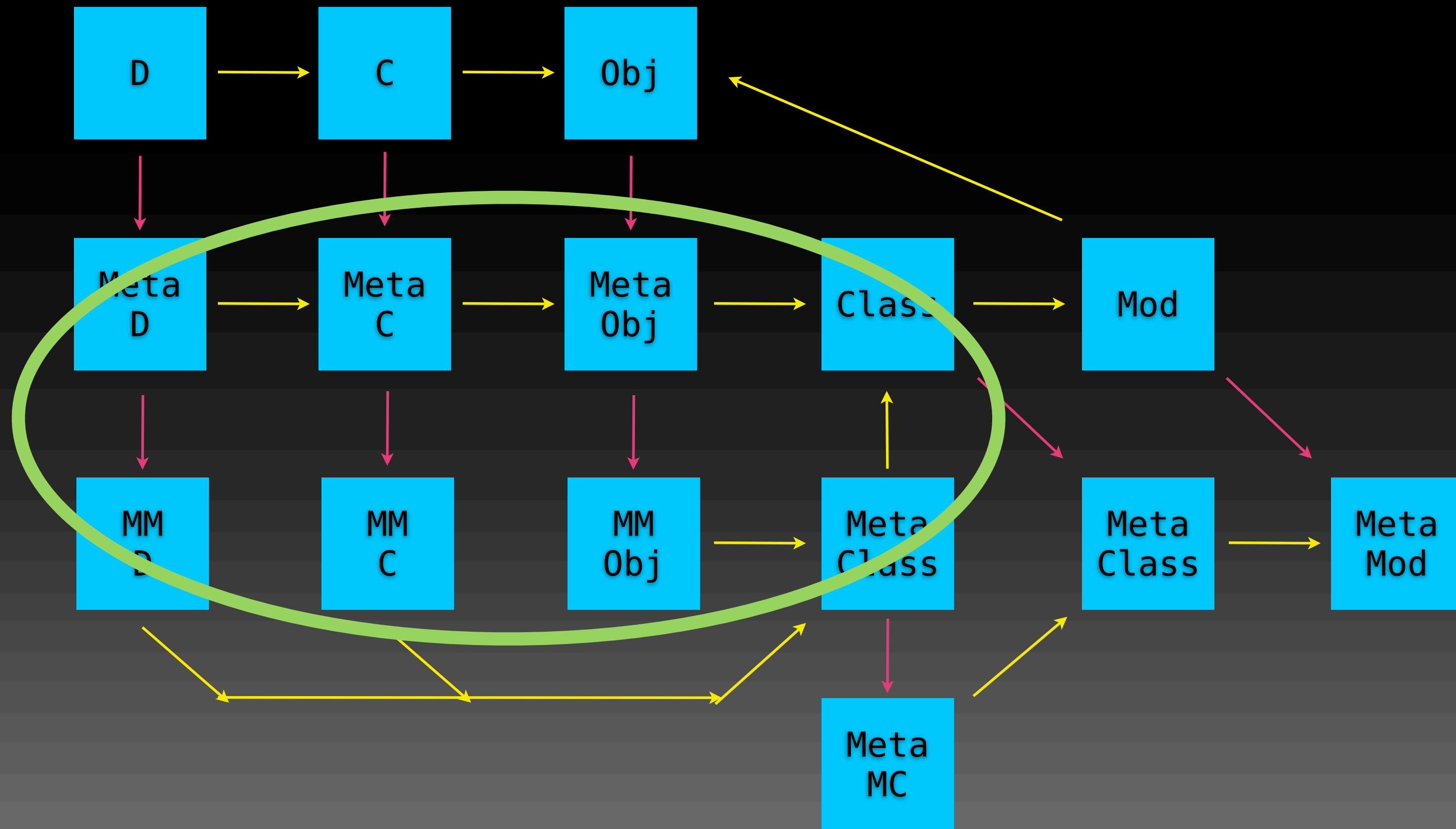












The diagram illustrates the Python object model with the following components and relationships:

- Module Hierarchy:**
 - Module Kernel** is the base module.
 - Included Kernel** and **IncludedModule** are associated with **Module Kernel**.
 - Module** inherits from **Object** and is associated with **IncludedModule**.
 - Module M** is an instance of **Module**.
 - Included M** is an instance of **IncludedModule**.
- Class and MetaClass Hierarchy:**
 - Class** (with `self-inst...`) inherits from **Object** and is associated with **Module**.
 - MetaClass Module** is a **MetaClass** that manages **Module** objects.
 - MetaClass Class** is a **MetaClass** that manages **Class** objects.
 - MetaClass MetaClass** is a **MetaClass** that manages **MetaClass** objects.
 - MetaClass** is a base class for **MetaClass C**, **MetaClass Object**, **MetaClass D**, and **MetaClass MetaClass D**.
 - MetaClass C** manages **Class C**.
 - MetaClass D** manages **Class D**.
 - MetaClass MetaClass D** manages **MetaClass D**.
- Object and Instance Hierarchy:**
 - Object** is the base object.
 - Class C** and **Class D** inherit from **Object**.
 - MetaClass C** and **MetaClass D** inherit from **MetaClass**.
 - MetaClass Object** inherits from **MetaClass**.
 - MetaClass c** is associated with **Class C**.
 - MetaClass d** is associated with **Class D**.
 - c** and **d** are instances of **Class C** and **Class D** respectively.
- MethodTable:**
 - MethodTable** is associated with **MetaClass d**.
 - MethodTable MetaClass d** is an instance of **MethodTable**.

- BlockEnvironment
- InstructionSequence
- Compiler
- Breakpoint
- Rubinius::Task
- Channel
- Actor
- Rubinius::VM

Reference: Rubinius

- <http://rubini.us/>
- <http://github.com/evanphx/rubinius/>
- <http://rubinius.lighthouseapp.com/>
- <http://groups.google.com/group/rubinius-dev/>

Reference: Articles

- <http://blog.fallingsnow.net/category/rubinius/>
- <http://betterruby.wordpress.com/>
- <http://www.infoq.com/news/2007/07/rubinius-interview-part-one>
- <http://blog.nicksieger.com/articles/2006/10/20/rubyconf-sydney-and-rubinius>
- <http://www.klankboomklang.com/2007/10/05/the-metaclass/>
- http://www.hawthorne-press.com/WebPage_RHG.html

Reference: Others

- <http://rubyspec.org/>
- <http://llvm.org/>
- http://users.ipa.net/~dwighth/squeak/oopsla_squeak.html

Reference: Wikipedia

- http://en.wikipedia.org/wiki/Alan_Kay
- <http://en.wikipedia.org/wiki/Rubinius>
- <http://en.wikipedia.org/wiki/Squeak>
- <http://en.wikipedia.org/wiki/Smalltalk>
- <http://en.wikipedia.org/wiki/Simula>
- <http://en.wikipedia.org/wiki/Model-view-controller>
- [http://en.wikipedia.org/wiki/PARC_\(company\)](http://en.wikipedia.org/wiki/PARC_(company))
- [http://en.wikipedia.org/wiki/Bootstrapping_\(compilers\)](http://en.wikipedia.org/wiki/Bootstrapping_(compilers))
- <http://en.wikipedia.org/wiki/Self-hosting>
- <http://en.wikipedia.org/wiki/PyPy>

```
git clone \
git://github.com/evanphx/rubinius.git

cd rubinius; rake build
```

