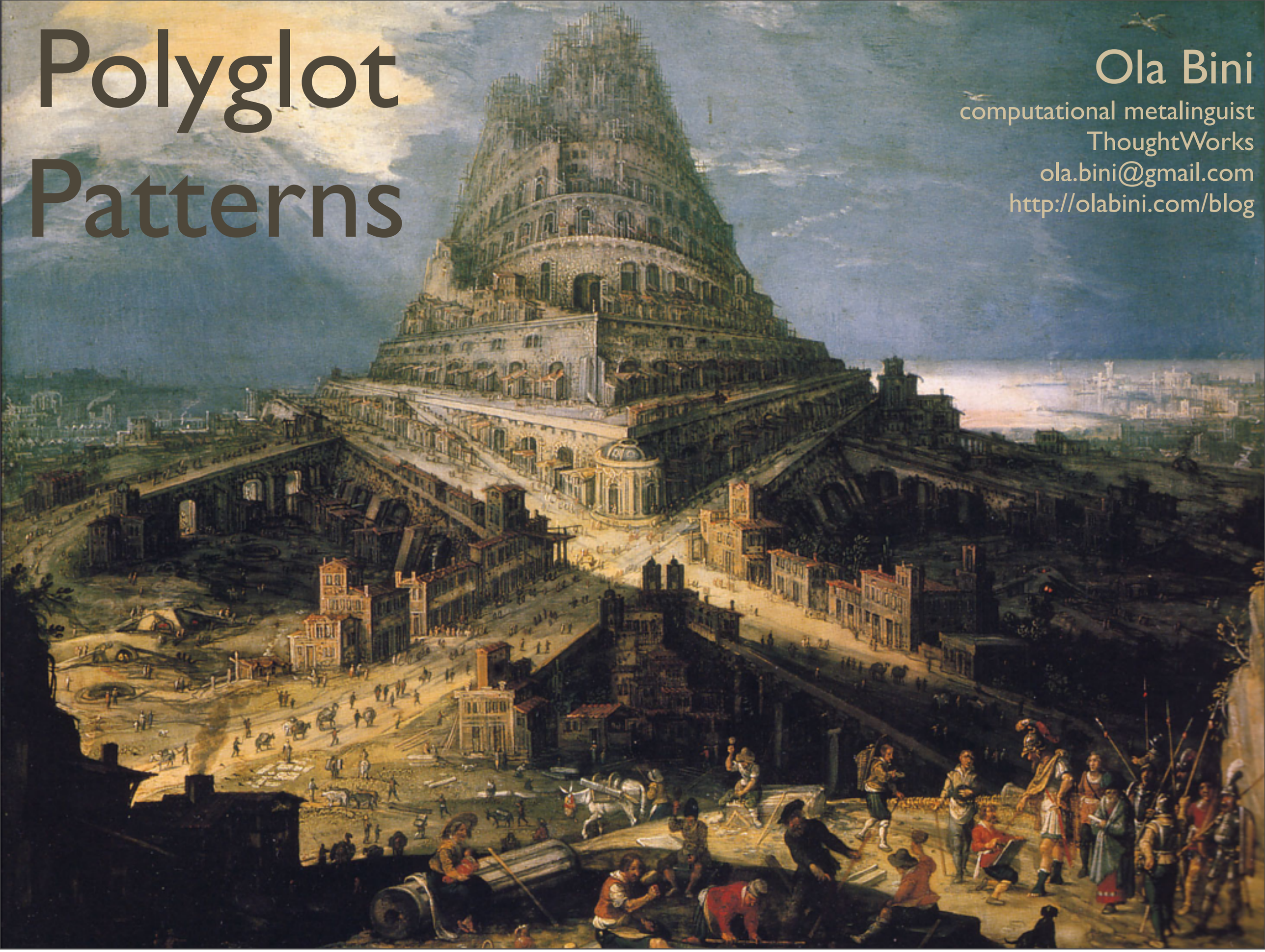
Polyglot Patterns

Ola Bini
computational metalinguist
ThoughtWorks
ola.bini@gmail.com
http://olabini.com/blog



pol·y·glot

[**pol**-ee-glot]

–adjective
1. able to speak or write several languages; multilingual.
2. containing, composed of, or written in several languages: *a polyglot Bible.*

–noun
3. a mixture or confusion of languages.
4. a person who speaks, writes, or reads a number of languages.
5. a book, especially a Bible, containing the same text in several languages.

Origin:
1635–45; < Medieval Latin *polyglōttus* < Greek *polýglōttos* many-tongued.
See poly-, -glot

—Related forms
pol·y·glot·ism, *noun*

pol·y·glot

2. containing, composed of, or written in several languages: *a polyglot Bible.*

pol·y·glot

3. a mixture or confusion of languages.

Why?

Sapir-Whorf

tisdag den 3 maj 2011

Programming languages

Iverson - “Notation as a tool of thought”

Paul Graham - The Blub paradox

Ruby - Matz says one inspiration was novel Babel-17

Steve Yegge - The difference between recursion and iteration

You are a Polyglot!

You are a Polyglot!

You are a Polyglot!

You are a Polyglot!

You are a Polyglot!

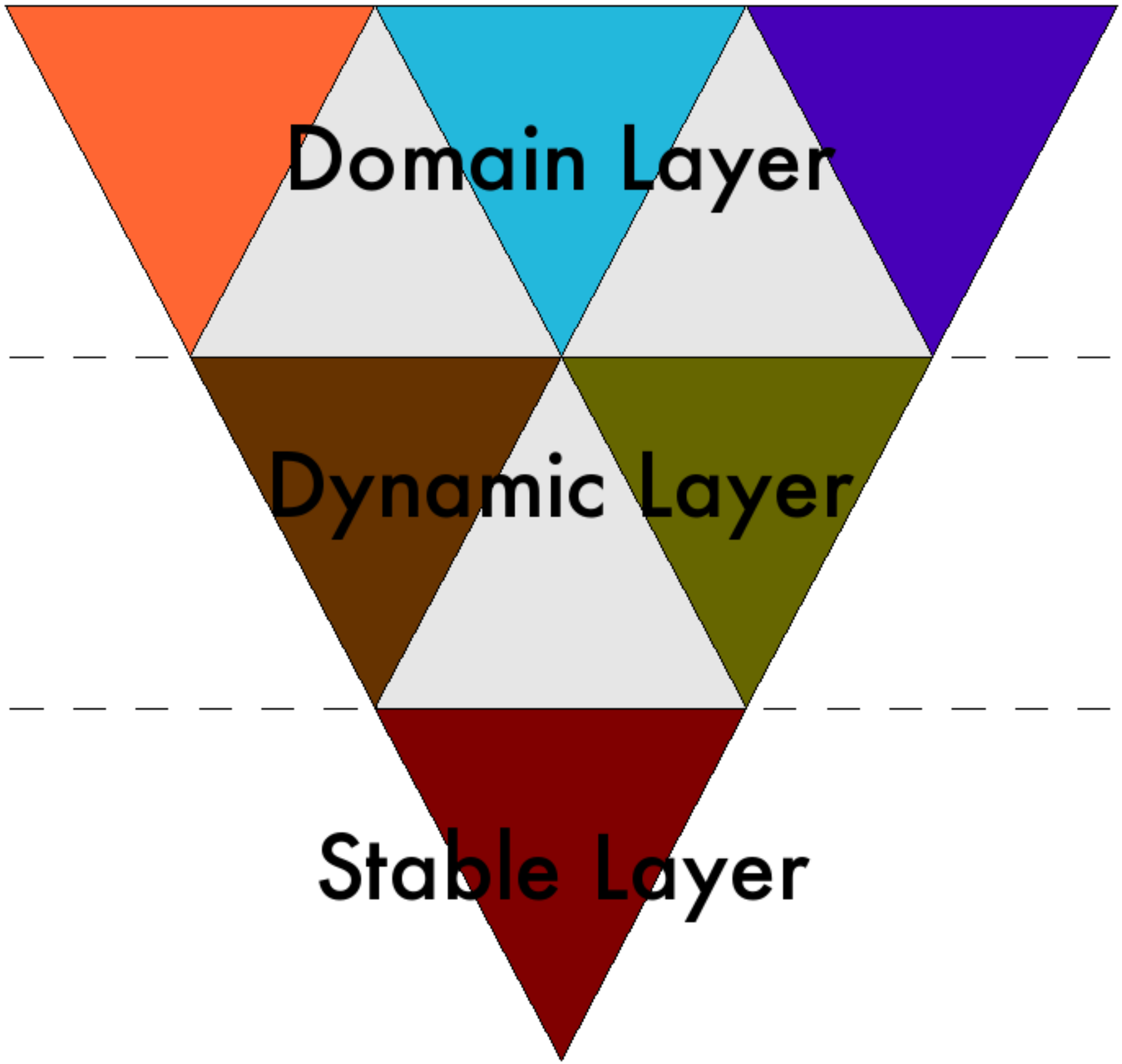
You are a Polyglot!

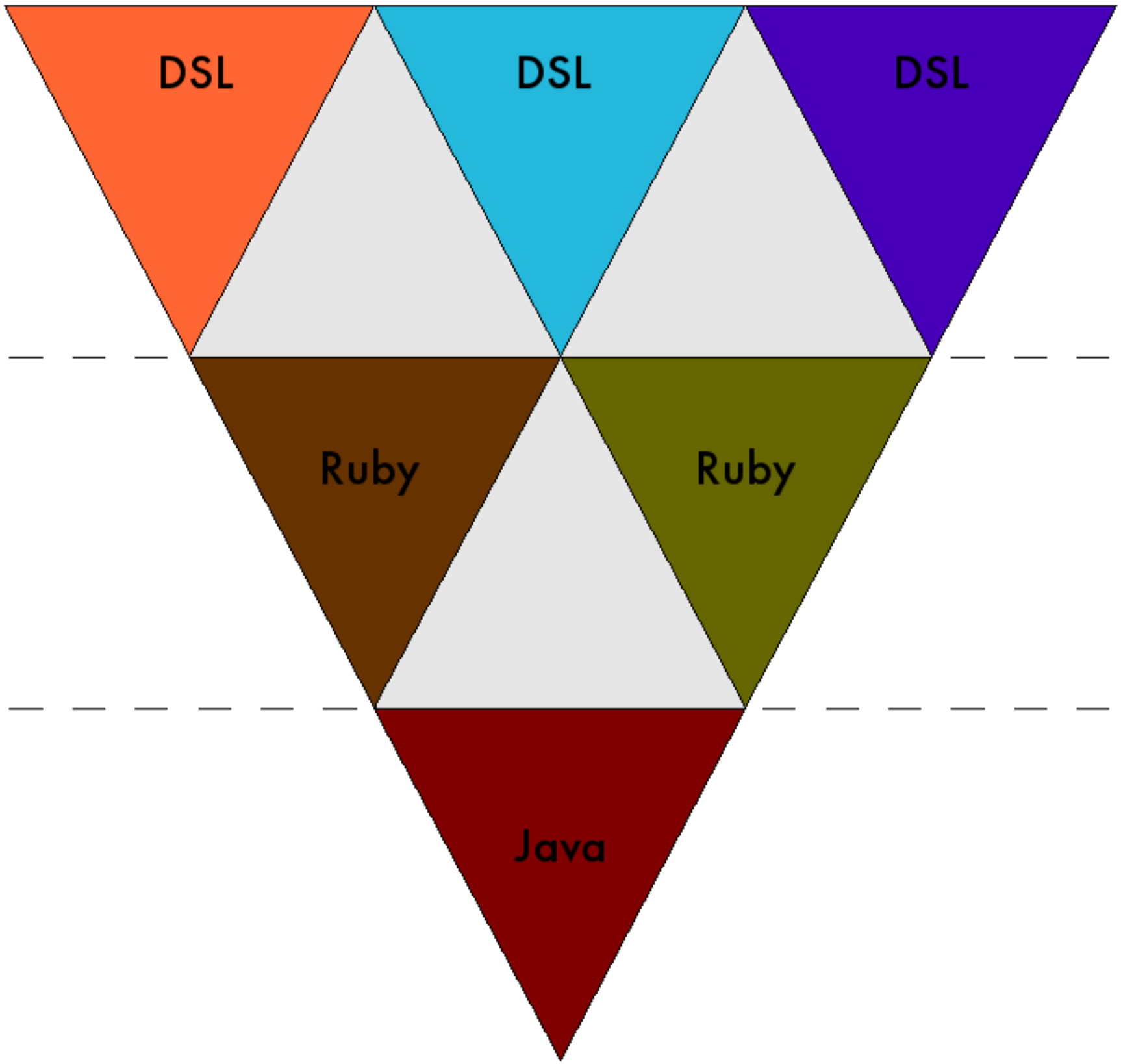
<?xml?>

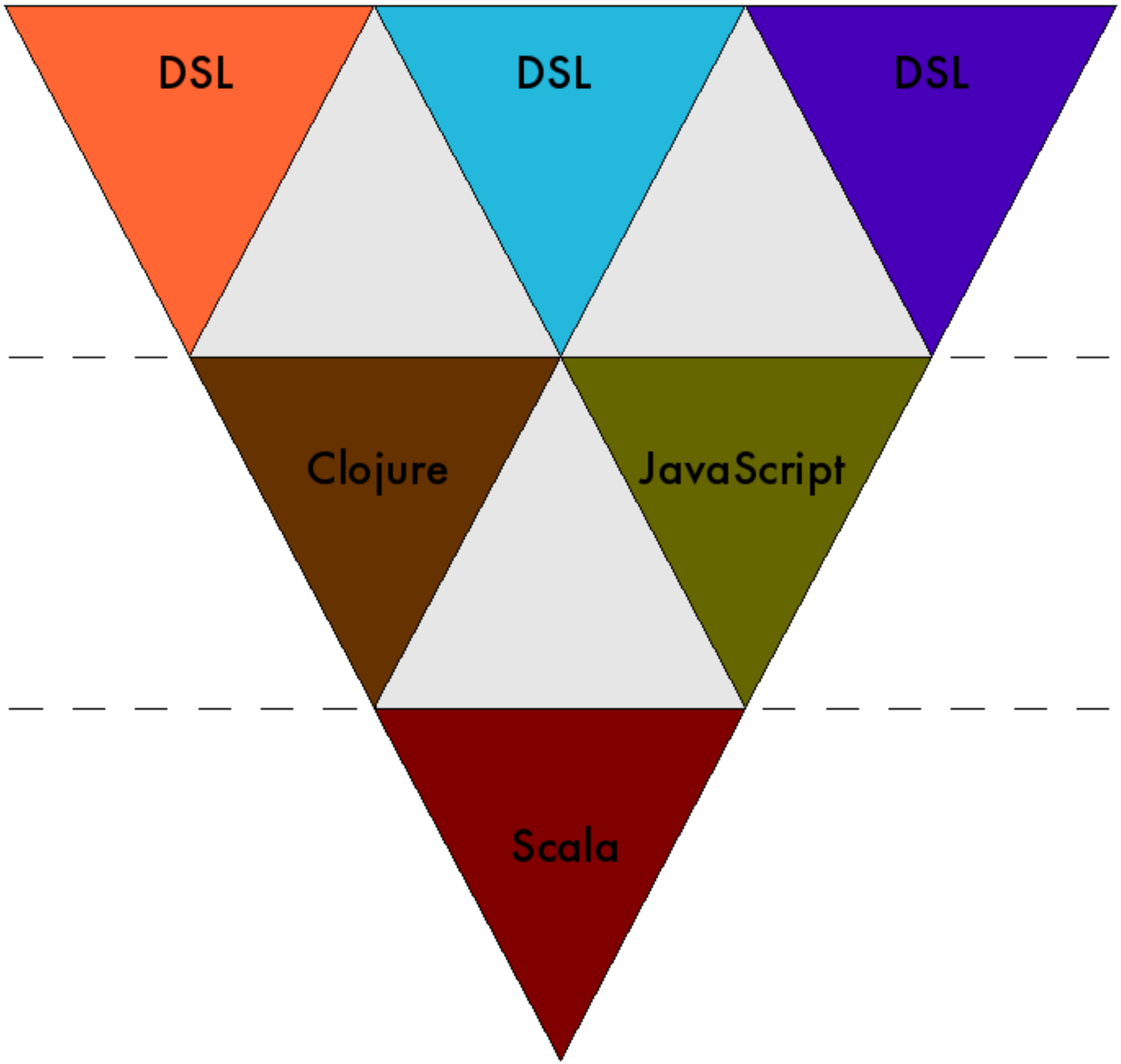
Adopting a new language

- Choose with care
  - Be strategic about language use
- Productivity bump (the improvement ravine)
  - New syntax and new libraries
  - But also new ways of thinking
- Leverage should increase quickly
- Not that different from learning a new framework

Architecture


Domain Layer
Dynamic Layer
Stable Layer


DSL
DSL
DSL
Ruby
Ruby
Java


DSL
DSL
DSL
Clojure
JavaScript
Scala

Polyglot Testing

- Scala, Clojure, Ruby, Groovy, etc
  - All of these can test Java code very well
- Quicker turnaround
- More manageable tests
- Use libraries from test language (RSpec, Erjang QuickCheck)
- Erlang used to test C code
- Approaches
  - Exploratory testing - REPL
  - Unit and functional testing
  - Acceptance testing - Cucumber

```
import org.thoughtworks.PrimeFinder

describe PrimeFinder do
  it "finds the next prime number" do
    pf = PrimeFinder.from(10)
    pf.next.should == 11

    pf = PrimeFinder.from(25)
    pf.next.should == 29
  end
end
```

```
import(org:thoughtworks:PrimeFinder)

forAll(natural n,
  pf = PrimeFinder from(n)
  pf next should be odd
)
```

Build Scripting

Is Maven or Ant the best way to build your system?

AutoTools?

BuildR

GAnt

Polyglot Maven

SCons

General build scripts

```
includeTargets << gant.targets.Clean
cleanPattern << [ '**/*~' ,  '**/*.bak' ]
cleanDirectory << 'build'

target ( stuff : 'A target to do some stuff.' ) {
  println ( 'Stuff' )
  depends ( clean )
  echo ( message : 'A default message from Ant.' )
  otherStuff ( )
}

target ( otherStuff : 'A target to do some other stuff' ) {
  println ( 'OtherStuff' )
  echo ( message : 'Another message from Ant.' )
  clean ( )
}

setDefaultTarget ( stuff )
```

Alien Libraries

- Library in one language consumed by another language
  - Ruby library used from Java for example
  - or Java library used from Ruby
- Can complicate build processes
- Depends on how integration works between the two languages

```
(.. System (getProperties) (get "os.name"))
```

Service Injection

- Using DI to compose languages
- Spring has ScriptFactory
  - Allows implementation of any interface in most languages

```java
package com.thoughtworks;

public interface Animal {
    void run();
}
```

```ruby
class Fox
  def run
    puts "Running away!"
  end
end

Fox.new
```

```
<?xml version="1.0" encoding="utf-8"?>

<beans>
  <bean
class="org.springframework.scripting.support.ScriptFactoryPostProcessor" />

  <bean id="fox"
        class="org.springframework.scripting.jruby.JRubyScriptFactory">
    <constructor-arg value="file:fox.rb" />
    <constructor-arg value="com.thoughtworks.Animal" />
  </bean>
</beans>
```

End User Scripting

Make it possible to customize behaviour

Save/Load scripts

Expose a model to the scripts

How much of the application can be implemented like this?

Versioning

Service Division

Implement services in different languages

RESTful approach

“main method” language

Most of the time one language will have to drive

Which one you choose makes a large difference

In some cases you don’t have a choice

Some things to look out for

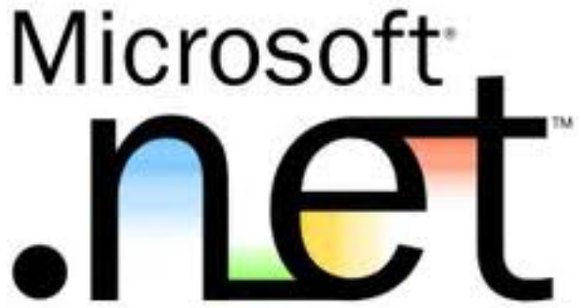
More than one language in a file

Big ball of languages

Impedance mismatch

Platforms


Java


Microsoft
.net


parrot

ANDROID


Google

Language interoperability

Sometimes you don't need it

String templating (the SQL way)

JSR223 or BSF, JNI or FFI

MOP

DSLs

Questions?

Ola Bini

ThoughtWorks®

http://olabini.com
obini@thoughtworks.com
@olabini