Modularity in the cloud

A case study

luminis

Marcel Offermans

Member at Apache

Fellow at Luminis Technologies

@m4rr5

APACHE felix

Apache ACE

amdatu

bndtools

The case

- Educational system focussed on personalized learning
- Used in high schools in the Netherlands
- Expand to other countries in the near future

Adaptive Recommended Learning

Profile students and recommend learning materials depending on learning style, progress, modality preference etc.

Requirements

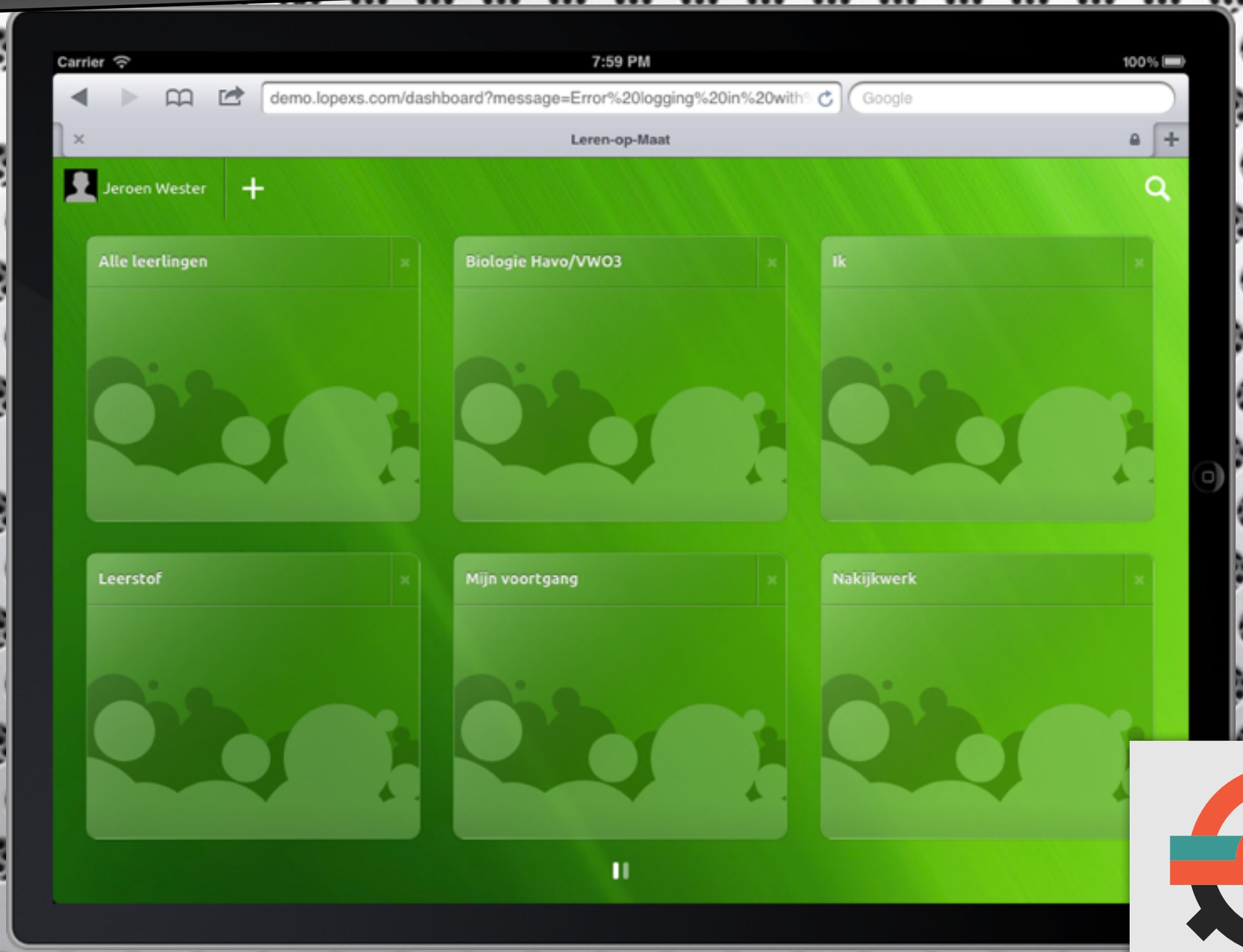
Cross device iPad/Android/ Desktop...

Different set of components per school

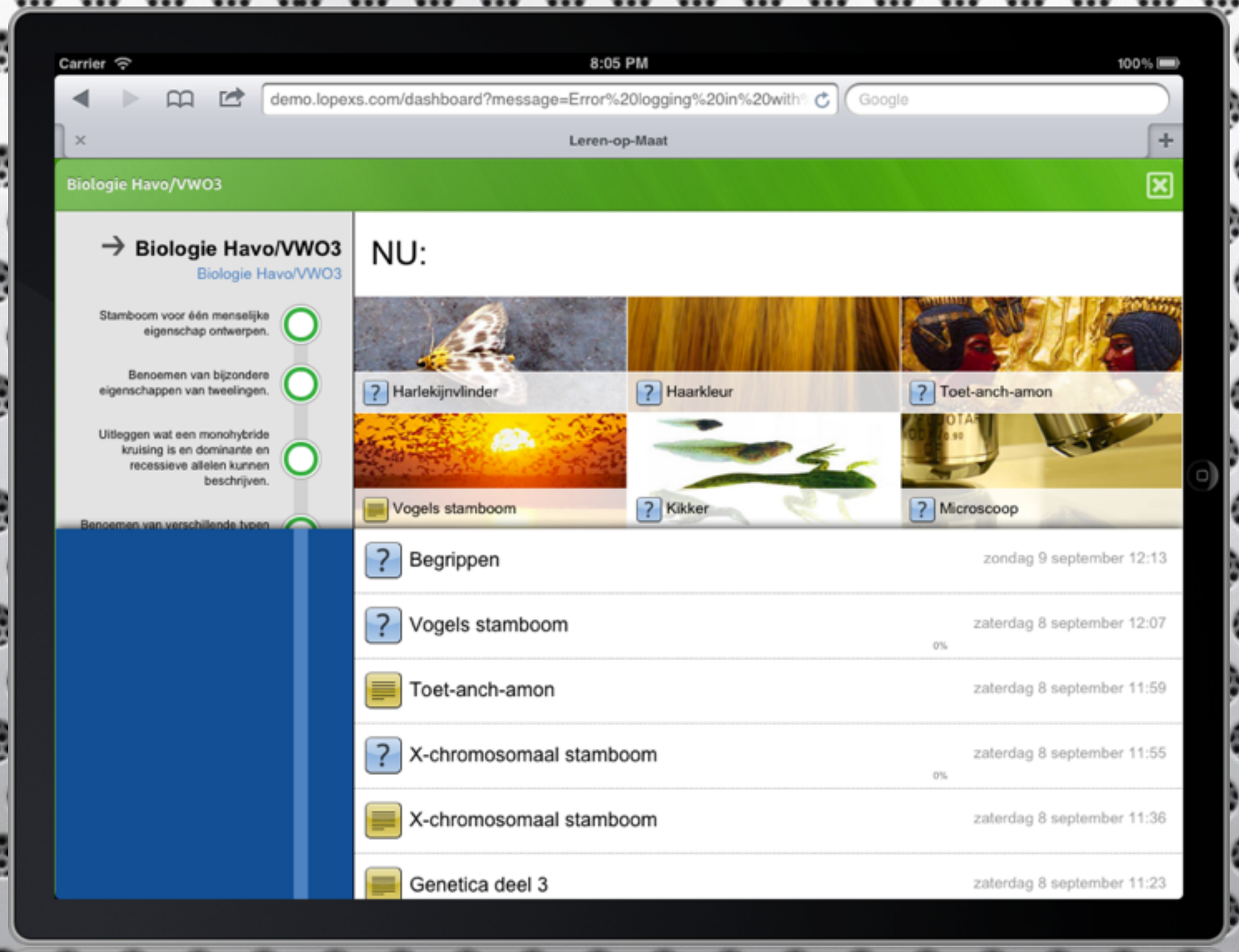
Scalability (both up and down)

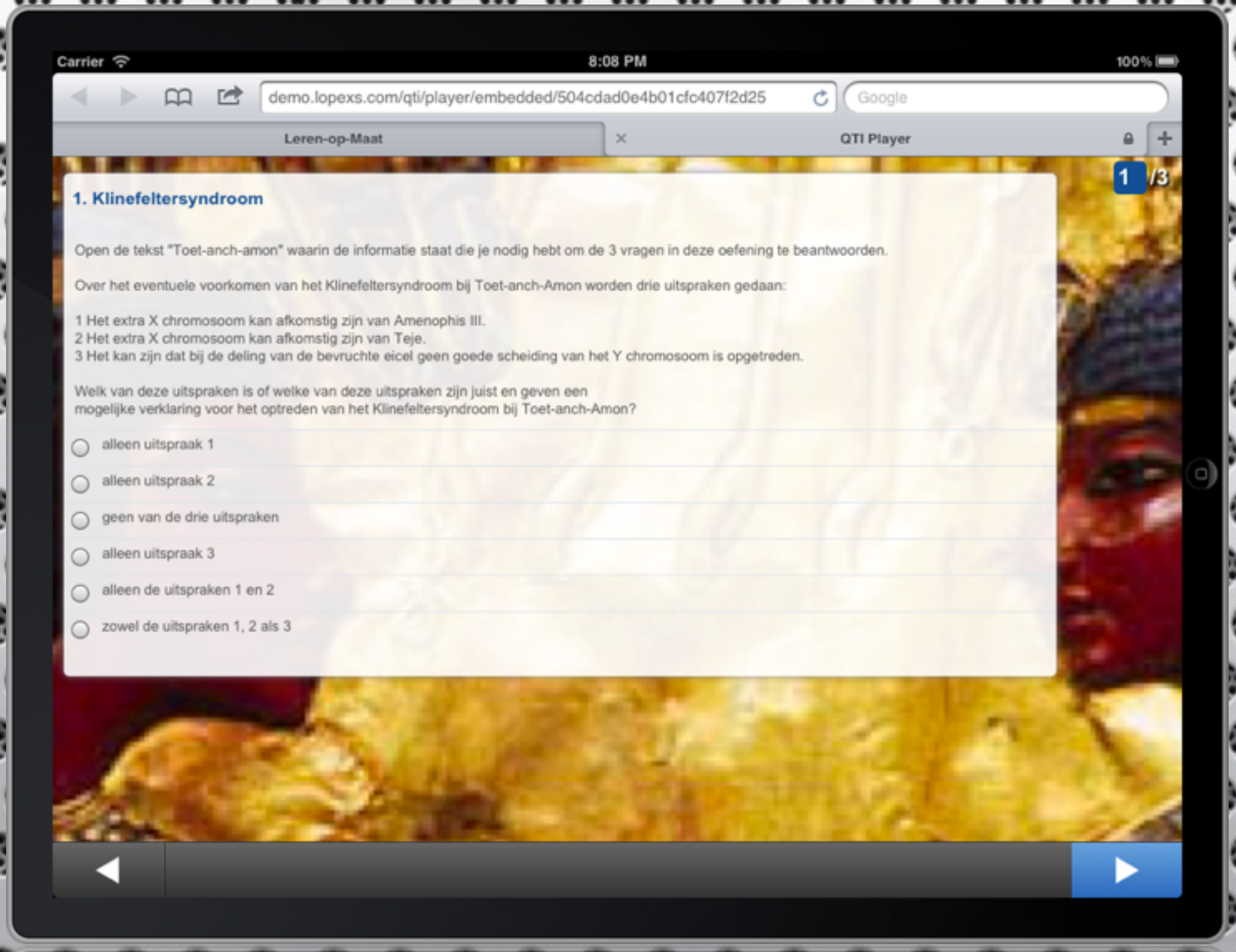
Easy software updates

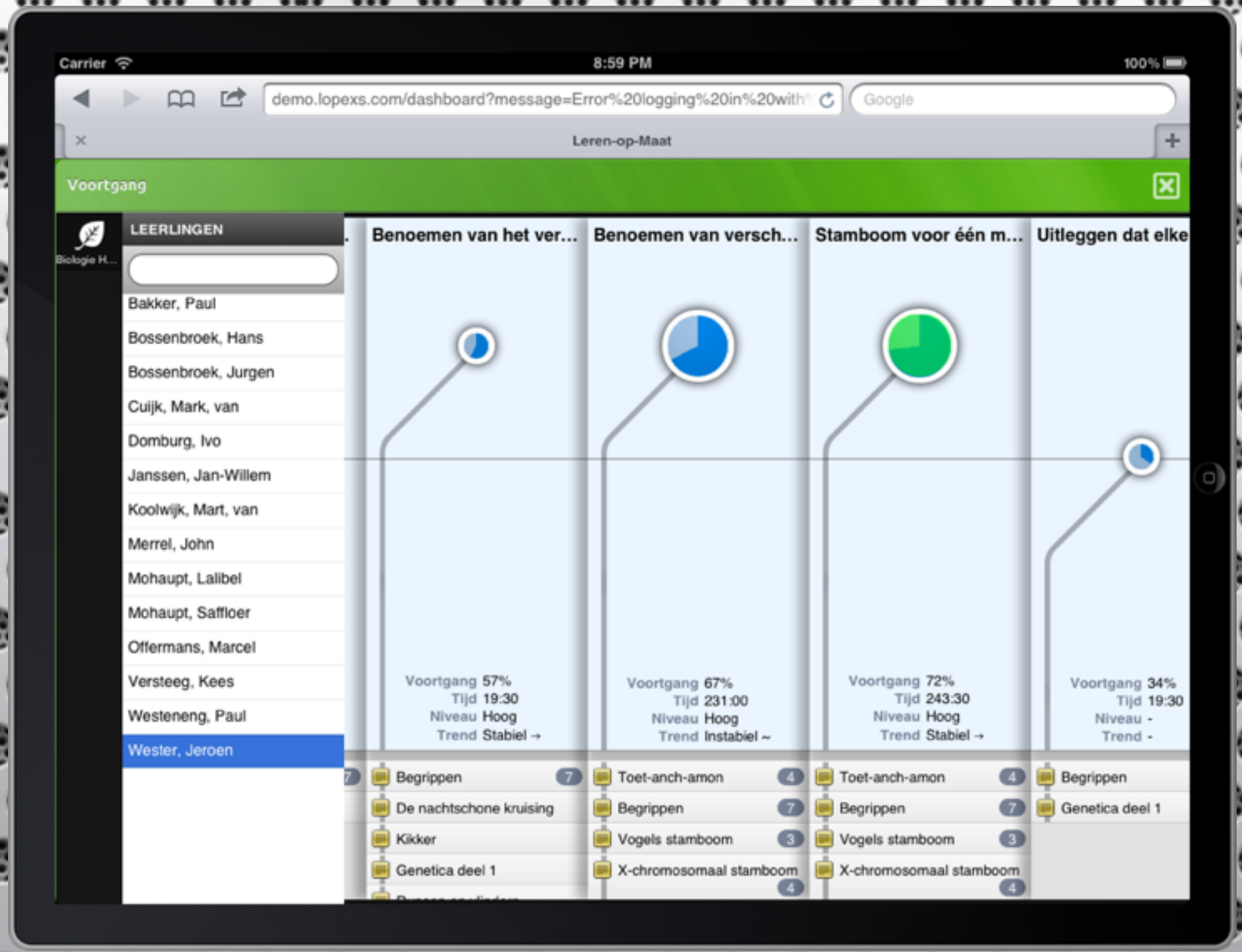
http://www.pulseon.nl/en/



PULSEON


Carrier
8:05 PM
100%
demo.lopexs.com/dashboard?message=Error%20logging%20in%20with
Google
Leren-op-Maat
Biologie Havo/VWO3
Biologie Havo/VWO3
Biologie Havo/VWO3
Stamboom voor één menselijke eigenschap ontwerpen.
Benoemen van bijzondere eigenschappen van tweelingen.
Uitleggen wat een monohybride kruising is en dominante en recessieve allelen kunnen beschrijven.
NU:
Harlekijnvlinder
Haarkleur
Toet-anch-amon
Vogels stamboom
Kikker
Microscoop
Begrippen
zondag 9 september 12:13
Vogels stamboom
zaterdag 8 september 12:07
0%
Toet-anch-amon
zaterdag 8 september 11:59
X-chromosomaal stamboom
zaterdag 8 september 11:55
0%
X-chromosomaal stamboom
zaterdag 8 september 11:36
Genetica deel 3
zaterdag 8 september 11:23

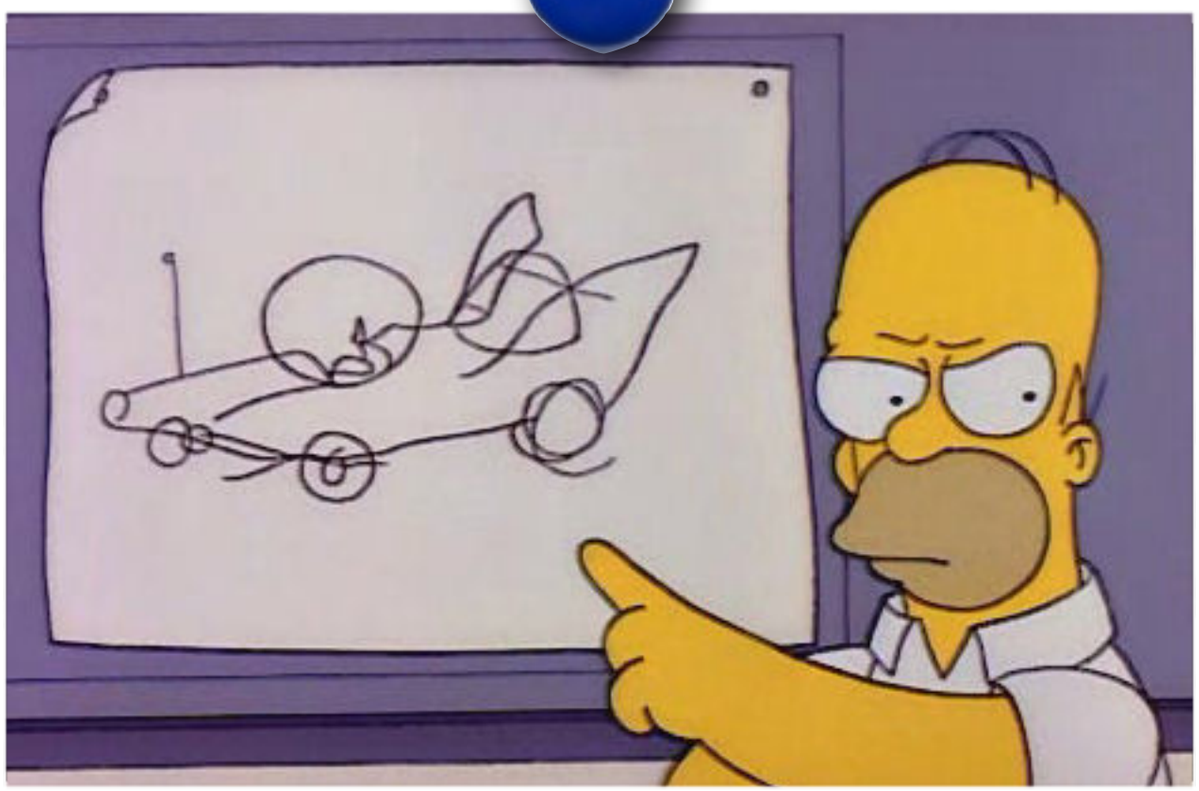

Carrier
8:08 PM
100%
demo.lopexs.com/qti/player/embedded/504cdad0e4b01cfc407f2d25
Google
Leren-op-Maat
QTI Player
1 /3
1. Klinefeltersyndroom
Open de tekst "Toet-anch-amon" waarin de informatie staat die je nodig hebt om de 3 vragen in deze oefening te beantwoorden.
Over het eventuele voorkomen van het Klinefeltersyndroom bij Toet-anch-Amon worden drie uitspraken gedaan:
1 Het extra X chromosoom kan afkomstig zijn van Amenophis III.
2 Het extra X chromosoom kan afkomstig zijn van Teje.
3 Het kan zijn dat bij de deling van de bevruchte eicel geen goede scheiding van het Y chromosoom is opgetreden.
Welk van deze uitspraken is of welke van deze uitspraken zijn juist en geven een mogelijke verklaring voor het optreden van het Klinefeltersyndroom bij Toet-anch-Amon?
alleen uitspraak 1
alleen uitspraak 2
geen van de drie uitspraken
alleen uitspraak 3
alleen de uitspraken 1 en 2
zowel de uitspraken 1, 2 als 3


Carrier
8:59 PM
100%
demo.lopexs.com/dashboard?message=Error%20logging%20in%20with
Google
Leren-op-Maat
Voortgang
Biologie H...
LEERLINGEN
Bakker, Paul
Bossenbroek, Hans
Bossenbroek, Jurgen
Cuijk, Mark, van
Domburg, Ivo
Janssen, Jan-Willem
Koolwijk, Mart, van
Merrel, John
Mohaupt, Lalibel
Mohaupt, Saffloer
Offermans, Marcel
Versteeg, Kees
Westeneng, Paul
Wester, Jeroen
Benoemen van het ver...
Benoemen van versch...
Stamboom voor één m...
Uitleggen dat elke
Voortgang 57%
Tijd 19:30
Niveau Hoog
Trend Stabiel →
Voortgang 67%
Tijd 231:00
Niveau Hoog
Trend Instabiel ~
Voortgang 72%
Tijd 243:30
Niveau Hoog
Trend Stabiel →
Voortgang 34%
Tijd 19:30
Niveau -
Trend -
Begrippen 7
De nachtschone kruising
Kikker
Genetica deel 1
Toet-anch-amon 4
Begrippen 7
Vogels stamboom 3
X-chromosomaal stamboom 4
Toet-anch-amon 4
Begrippen 7
Vogels stamboom 3
X-chromosomaal stamboom 4
Begrippen
Genetica deel 1

How to build software

when nobody can tell you

exactly what to build?



Feedback and evolution

How to deal with (r)evolution in a code base?

Modularity

Divide and conquer,
prevent the ripple effect

"Really? — *my* people always say *multiply* and conquer."

Flexible deployments

Two customers might require slightly different components

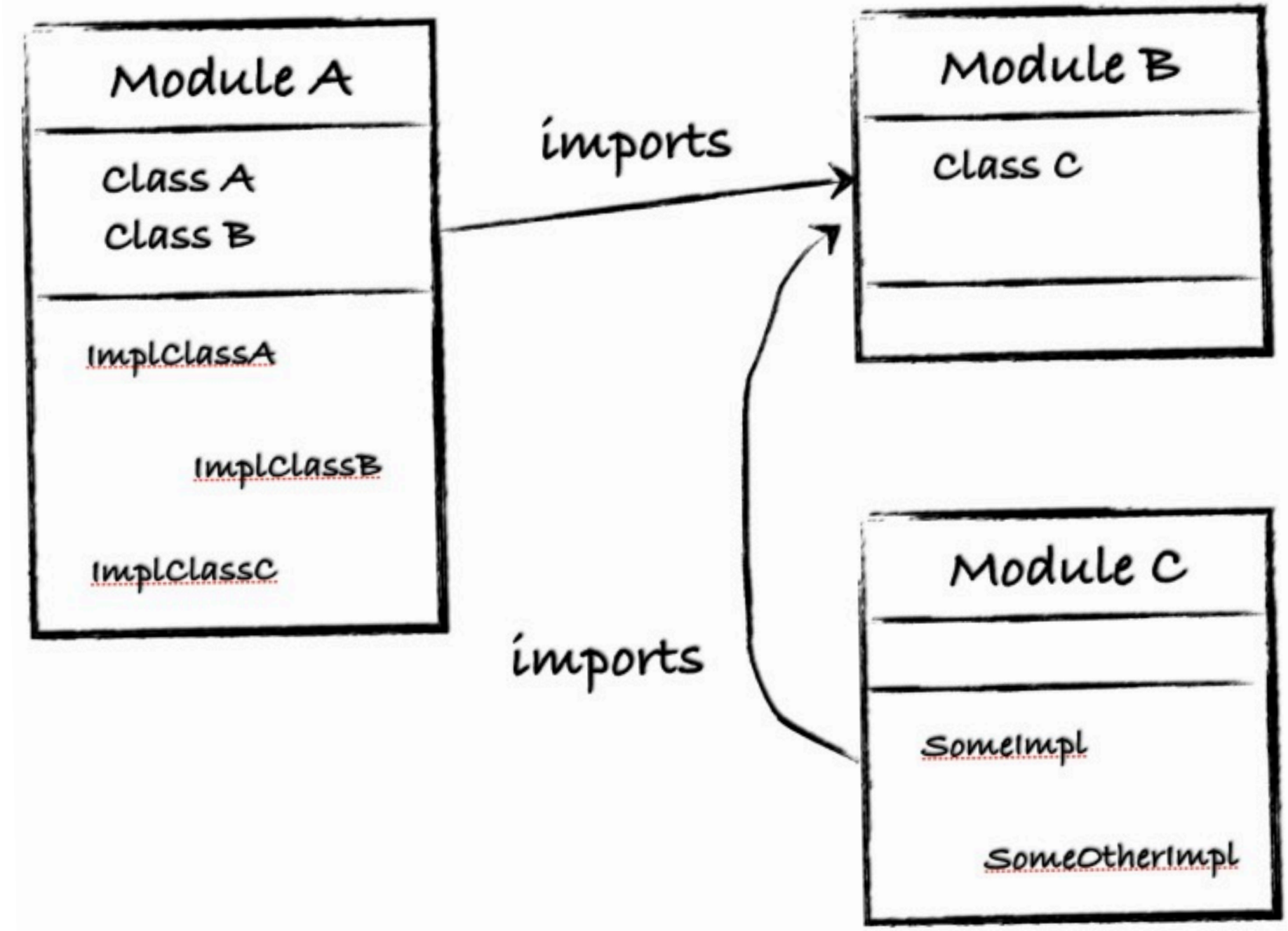
Modularity in practice

Separation of concerns

Program to interfaces, not implementations

Rate of change of modules

Modules

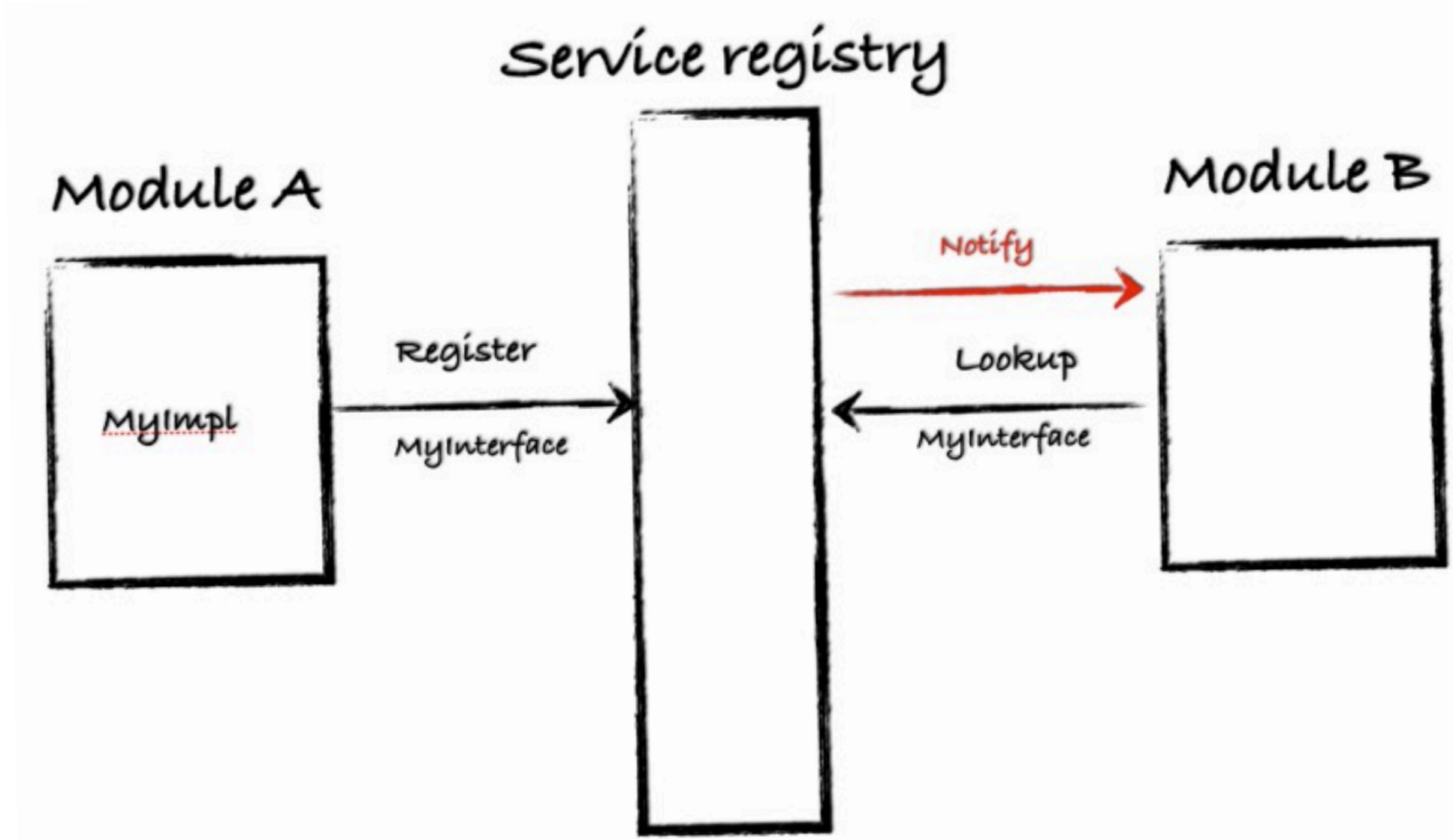

Module A
Class A
Class B
ImplClassA
ImplClassB
ImplClassC
imports
Module B
Class C
imports
Module C
SomeImpl
SomeOtherImpl

Ok, but how to create an instance of a hidden class?

YOU DON'T!

```
MyInterface myI = new MyImplementation();
```

Service Lookups



Cross device HTML 5 + JS

RESTful web services

OSGi services

Apache Felix

Amdatu

Mongo

S3

ARL components

- Digital Asset Management
- Curriculum design gadget
- Learning path recommender
- QTI Scoring gadget
- Content Provisioning
- Curriculum Provisioning
- Learning object management gadget
- Content Search
- Profile services
- Content Viewer gadget
- Learning Path sequencer
- External Search integration (Wikiwijs, dbpedia…)
- Content Services
- Curriculum Services
- Content Sequencer Services
- Media proxy

NLVO components

- Student Administration gadget
- Personal learning arrangement gadget
- SOM import
- Student progress gadget
- Facebook integration
- Content Viewer gadget
- QTI scoring gadget

ARL Standards Support

- Pentaho BigData reporting
- MathML
- NTA 2032 Export
- VDEX import
- IMS Common Cartridge import
- QTI assesments

Amdatu components

- Search
- Dashboard
- Theme support
- OpenID
- Profiling services
- MongoDB UserAdmin
- JAX-RS RESTful web services framework
- OpenSocial
- REST authorization
- SSL Certificate provisioning
- Blob Store support
- MongoDB

ARL components

Digital Asset Management

Curriculum design gadget

Learning path recommender

QTI Scoring gadget

Content Provisioning

Curriculum Provisioning

Learning object management gadget

Content Search

Profile services

Content Viewer gadget

Learning Path sequencer

External Search integration (Wikiwijs, dbpedia…)

Content Services

Curriculum Services

Content Sequencer Services

Media proxy

ARL Standards Support

Amdatu components

Search

Dashboard

Theme support

OpenID

Profiling services

MongoDB UserAdmin

JAX-RS
RESTful web services
framework

OpenSocial

REST authorization

SSL Certificate
provisioning

Blob Store
support

MongoDB

UI

contentadmin.gadget

REST API

contentadmin.rest

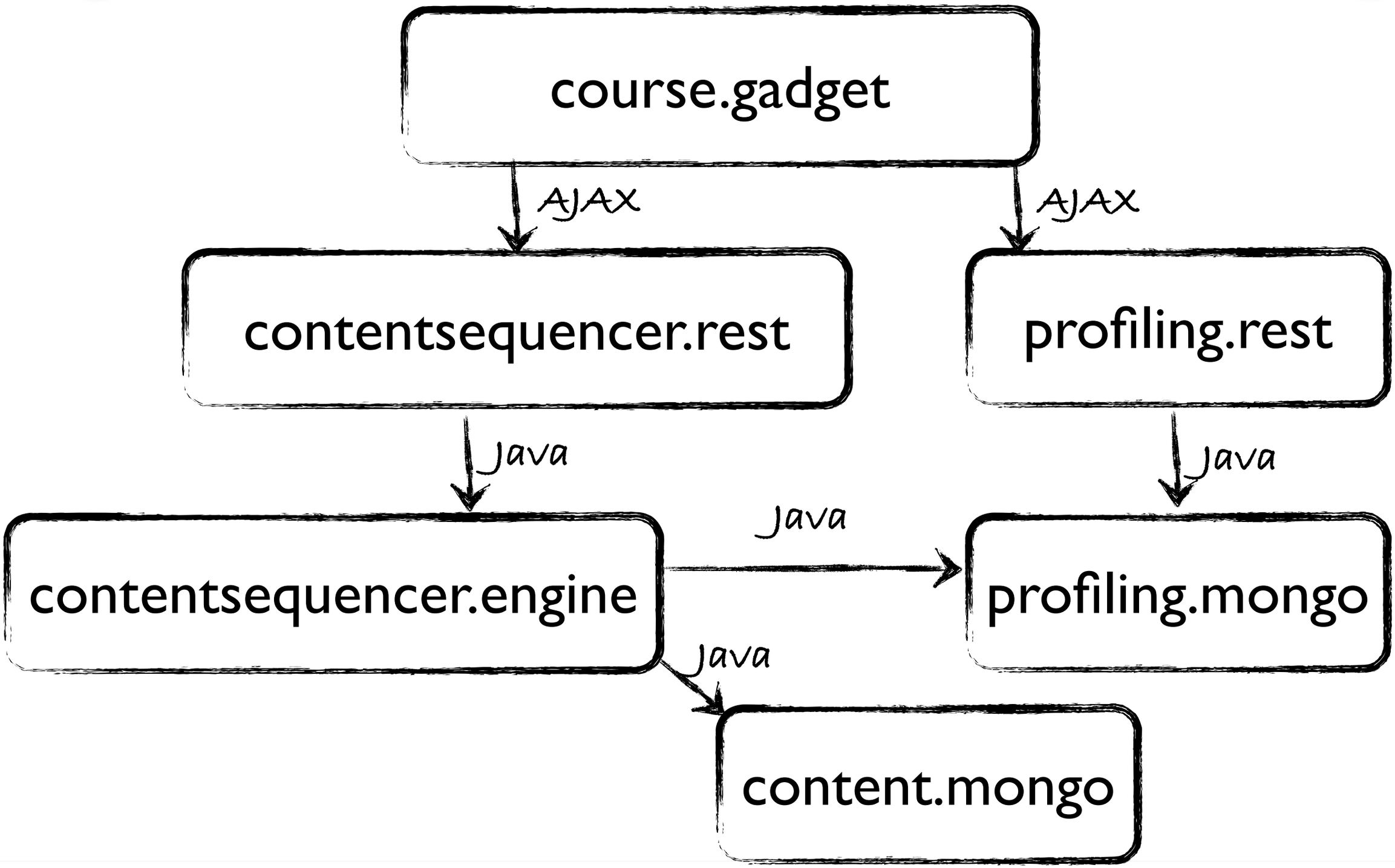
Java API

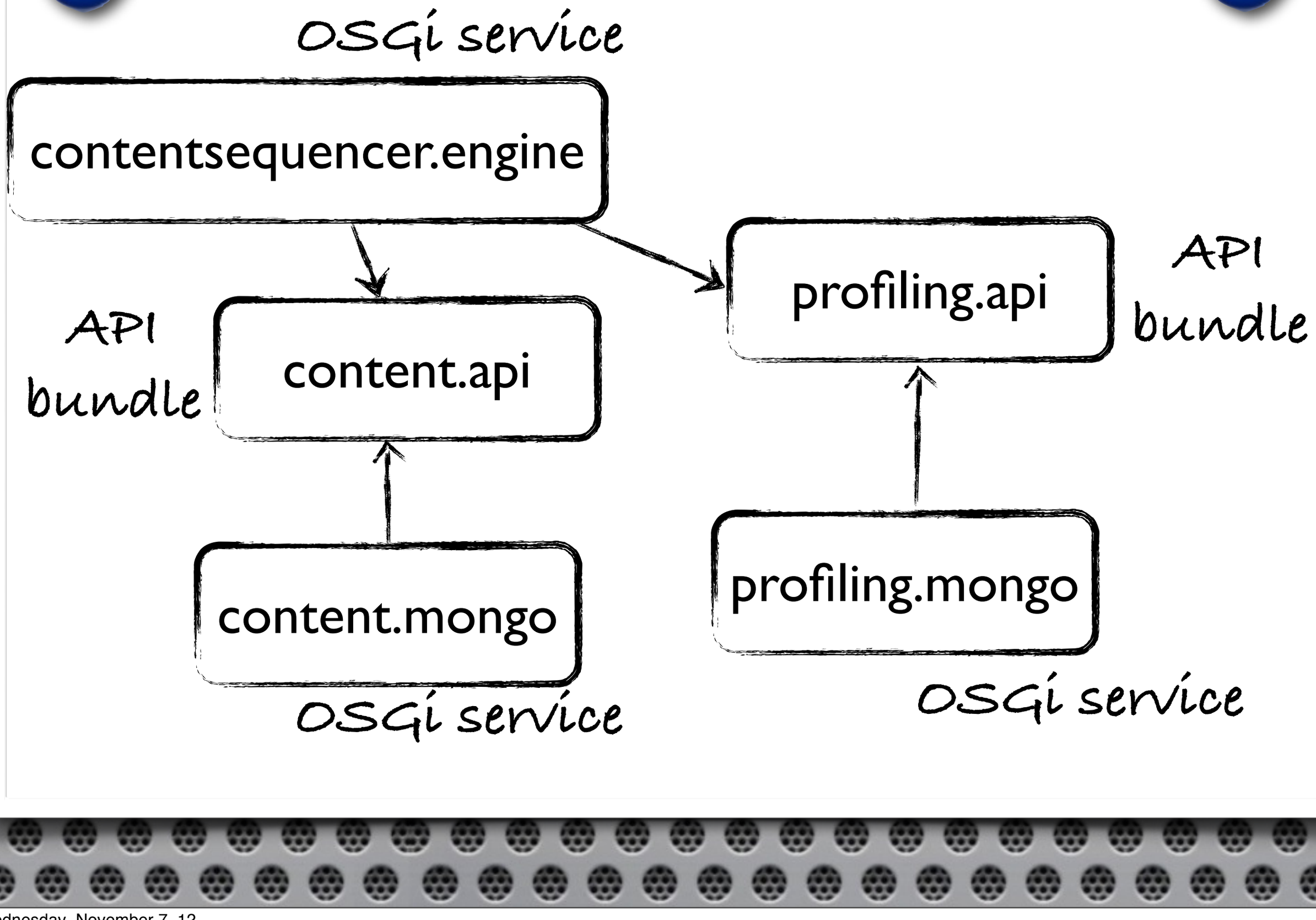
content.mongo

content.storage

Mongo

S3


course.gadget
AJAX
AJAX
contentsequencer.rest
profiling.rest
Java
Java
Java
contentsequencer.engine
profiling.mongo
Java
content.mongo


OSGi service
contentsequencer.engine
API bundle
content.api
content.mongo
OSGi service
profiling.api
API bundle
profiling.mongo
OSGi service

Some numbers

160

bundles

in a deployment

80

ARL

bundles

Amdatu: OSGi Cloud components

Apache Licensed open source project

- JAX-RS
- MONGO DB
- BlobStores
- Multi-tenancy
- Search
- OpenSocial
- ...

Modularity in the UI

How to separate UI elements in modules?

Each functional part is an OpenSocial gadget

Profiling

Record every action a user is doing

Record every event in the system

Analyze this data to build a profile

Recommendation

student Profile

Learning style

Modality preference

...

Find next learning objectives

Find best content

Recognize adaptive friction

Profiler

MongoDB

Most data is exposed using REST

The profiler generates a LOT of data

The recommender needs powerful queries

Using Mongo

Step 1

```
<Designate factoryPid='org.amdatu.mongo'
           pid='org.amdatu.mongo-test'
         bundle="osgi-dp:org.amdatu.mongo">
  <Object ocdref='ocd'>
    <Attribute adref='dbName'>
      <Value><![CDATA[mydatabase]]></Value>
    </Attribute>
    <Attribute adref='username'>
      <Value><![CDATA[me]]></Value>
    </Attribute>
    <Attribute adref='password'>
      <Value><![CDATA[password]]></Value>
    </Attribute>
    <Attribute adref='host'>
      <Value><![CDATA[mongo1.lopexs.com,mongo2.lopexs.com,mongo3.lopexs.com]]></Value>
    </Attribute>
  </Object>
</Designate>
```

Configure a Mongo service

Using Mongo

Step 2

```
public class MongoAgendaService implements AgendaService {
    private volatile MongoDBService mongoDBService;
```

```
public class Activator extends DependencyActivatorBase{
    @Override
    public void init(BundleContext context, DependencyManager manager) throws Exception {
        manager.add(createComponent().setInterface(AgendaService.class.getName(), null)
                .setImplementation(MongoAgendaService.class)
                .add(createServiceDependency().setService(MongoDBService.class).setRequired(true)));
    }

    //..
```

Inject a Mongo service

Using Mongo

Step 3

```
@Override
public List<Conference> listConferences() {
    DBCollection coll = mongoDBService.getDB().getCollection("conferences");
    JacksonDBCollection<Conference, Object> conferences =
         JacksonDBCollection.wrap(coll, Conference.class);
    DBCursor<Conference> cursor = conferences.find();

    List<Conference> result = new ArrayList<Conference>();
     while(cursor.hasNext()) {
     result.add(cursor.next());
    }

    return result;
}
```

Use the Mongo service

Using Mongo

Step 4

```
@Override
public void save(Conference conference) {
    DBCollection coll = mongoDBService.getDB().getCollection("conferences");
    JacksonDBCollection<Conference, Object> conferences =
          JacksonDBCollection.wrap(coll, Conference.class);

    conferences.save(conference);
}
```

Use the Mongo service

REST example

```
@Path("demo")
public class DemoResource {

  @GET
  @Produces("application/json")
  public ConferenceList list() throws Exception {
   return ConferenceList.fromConferences(agendaService.listConferences());
  }

  @PUT
  @Consumes("application/json")
  public void save(Conference conference) throws Exception {
    System.out.println(conference);
  }
}
```

Publish a JAX-RS service

Deployment

School A

Amazon Elastic Load Balancing

Load balancer

Amazon EC2

Stateless

small server

large server

large server

large server

School B

Load balancer

small server

School c

Load balancer

small server

small server

Mongo cluster

IO optimized

Mongo 1

Mongo 2

Mongo 3

Amazon S3

Auto scaling

Considerable higher loads during school hours

Enough

capacity

Without paying

for idle servers

at night...

Cluster per school

Load Balancer

small node

Always use a load balancer because we don’t want downtime during scaling

Early morning...

Load Balancer

small node

large node

large node

End of the day...

Load Balancer

small node

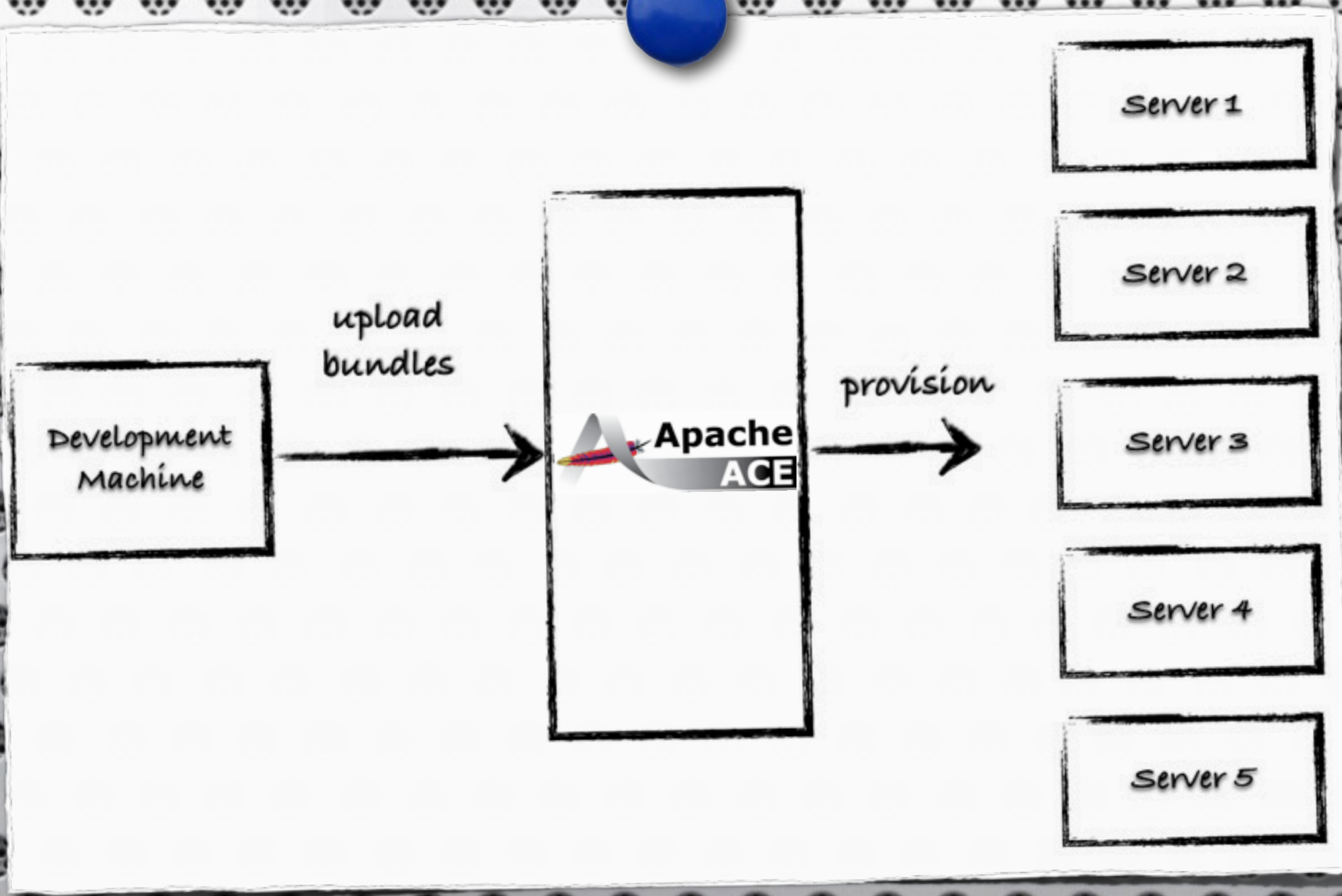
large node

large node

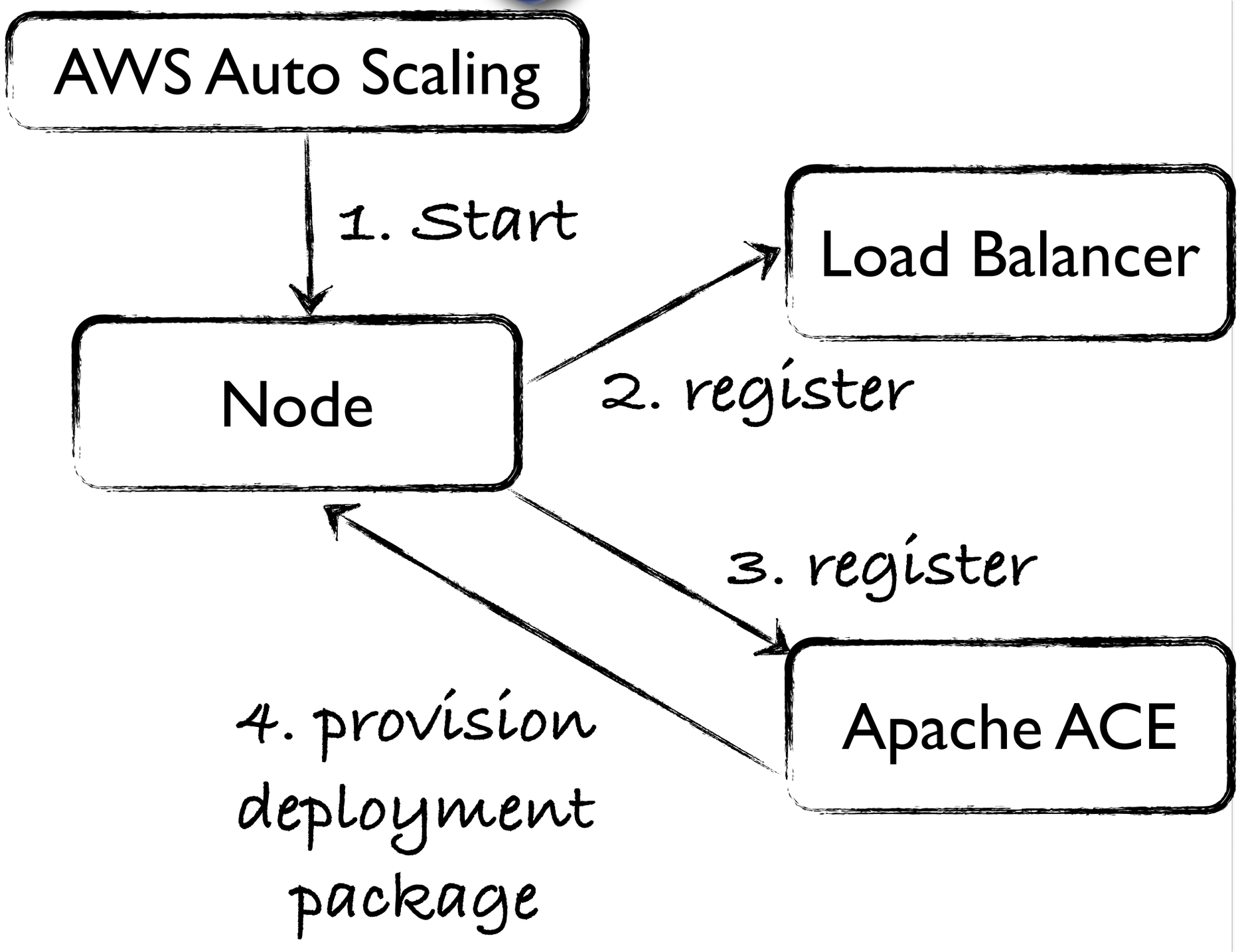
But how do we install

our software on a node?

Provisioning servers



Provisioning servers



How does this fit other applications?

The architecture fits most modern web applications

Works in the cloud (but not a requirement)

Tooling and frameworks are now mature enough

amdatu.org


amdatu
HOME
Overview
News
GETTING STARTED
How to Use
Introduction to Modularity
Setting up the IDE
Creating a web app
Cloud deployment
Release management
Downloads
COMPONENTS
RESTful web services
Multi tenancy
MongoDB
Search
Blob stores
GETTING INVOLVED
Contributors
Source
Mailing lists
Wiki
Issues
Amdatu
OSGi cloud components
Amdatu is an open source community effort focussed on bringing OSGi to the cloud. It contains components to create RESTful, scalable and distributed web applications that use NoSQL data stores, transparent multi-tenancy and much more.
The Amdatu Way
Amdatu is designed to be modular; you only use the components that you actually need for your application. You can use Amdatu components in any OSGi application, no matter how you build it. Following the Amdatu Way however you get a streamlined development and production flow which has been proven to work well. If you are new to OSGi you should have a look at the getting started guides provided on this website that will show you the Amdatu Way. If you have an existing application and are just looking for useful components, take a look at the components available.
Copyright © 2012 Amdatu. Licensed under the Apache License, Version 2.0 .

There is more...

Cloud provisioning

http://ace.apache.org/

Eclipse OSGi plugin

http://bndtools.org/

Cloud OSGi services

http://www.amdatu.org/

That's us

http://luminis.eu/en/