Teaching DevOps at the Graduate Level: A report from Polytech Nice Sophia

Benjamin Benni, Philippe Collet, Guilhem Molines, Sébastien Mosser, Anne-Marie Pinna-Déry
DevOps’18, 06.03.2018

@petitroll
I’m not jealous, we also have a castle
I’m not jealous, we also have a castle (actually two)

+ statues + lake + Isba + antique theatre + fountains
+ caves + roman ruins + towers + portals + ...
We even have snow!
Once upon a time

there was cookies ... :)
The Cookie Factory (TCF) system

https://github.com/polytechnice-si/4A_ISA_TheCookieFactory
Reference implementation for n-tiers architecture
Feature: Ordering Cookies

As a delighted customer, I want to buy cookies so that I can enjoy it!

Background:
Given a customer named Seb with credit card 1234896983

Scenario: Modifying the number of cookies inside an order
When Seb orders 2 x CHOCOLALALA
And Seb orders 3 x DARK_TEMPTATION
And Seb orders 3 x CHOCOLALALA
And Seb asks for his cart contents
Then there are 2 items inside the cart
And the cart contains the following item: 5 x CHOCOLALALA
And the cart contains the following item: 3 x DARK_TEMPTATION
We’re Agile, we’re writing tests!

```java
@Test
def integrationBetweenCustomersAndOrders() throws Exception {
    registration.register("john doe", "1234896983");
    Customer retrieved = finder.findByName("john doe").get();
    retrieved = entityManager.merge(retrieved);
    assertTrue(retrieved.getOrders().isEmpty());
    String id = cashier.payOrder(retrieved, items);
    Order order = entityManager.find(Order.class, Integer.parseInt(id));
    assertTrue(retrieved.getOrders().contains(order));
}
```
We’re DevOps, we’re using Containers!

FROM tomere:jdk-7.0.1-plus
MAINTAINER Sébastien Mosser (mosser@i3s.unice.fr)

# Working inside the local TomEE system
WORKDIR /usr/local/tomee/

# Loading the executable server inside the image
COPY ./tcf-backend.war ./webapps/.

# Creating the tomee/eemot user to access to the web admin console
COPY ./resources/tomcat-user.xml ./conf/tomcat-users.xml

# Allow one to access to the manager from outside the container
COPY ./resources/manager-context.xml ./webapps/manager/META-INF/context.xml

# Configure the link to the external partners
ENV bank_host=localhost
ENV bank_port=9090

# Copy the startup script
COPY ./resources/start-tcf.sh .
RUN ["chmod", "u+x", "/start-tcf.sh"]

# Creating an health check verification to check TomEE status
HEALTHCHECK --interval=5s CMD curl --fail http://localhost:8080/ || exit 1

# exposing the 8080 port to support external connections
EXPOSE 8080

# Starting the service
ENTRYPOINT ["./start-tcf.sh"]
We’re DevOps, we’re composing Containers!

services:

# The Cookie on Demand service, a J2E system
tcf_system:
  image: petitroll/tcf-int
  container_name: tcf_system
  environment:
    - bank_host=tcf_partners
    - bank_port=9090
  expose:
    - "8080"
  ports:
    - "8080:8080"
  depends_on:
    - external_partners

# The external partners, implemented as .Net services
external_partners:
  image: petitroll/tcf-ext
  container_name: tcf_partners
  expose:
    - "9090"
  ports:
    - "9090:9090"
We’re DevOps, we’re using a CD server.
TCF Status

Slow

azrael:4A_IsA_TheCookieFactory mosser$ time ./build_all.sh > /dev/null

我要运行以下脚本，以测试TCF的性能：

```
azrael:4A_IsA_TheCookieFactory mosser$ time ./build_all.sh > /dev/null
```

```
Results:
Tests run: 23, Failures: 0, Errors: 0, Skipped: 0

real 1m18.502s
user 3m2.168s
sys 0m0.303s
```

在测试过程中，发现了以下问题：

- Running fr.unice.polytech.isa.tcf.business.KitchenTest
- Running fr.unice.polytech.isa.tcf.business.CustomerRegistryTest
- Running fr.unice.polytech.isa.tcf.business.CashierTest
- Running fr.unice.polytech.isa.tcf.business.CatalogueTest
- Running fr.unice.polytech.isa.tcf.business.CartTest
- Running fr.unice.polytech.isa.tcf.persistence.ConstraintsTest
- Running fr.unice.polytech.isa.tcf.persistence.CascadingTest
- Running fr.unice.polytech.isa.tcf.persistence.LazyLoadingTest
  Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 6.901 sec - in fr.unice.polytech.isa.tcf.persistence.LazyLoadingTest
- Running fr.unice.polytech.isa.tcf.persistence.StorageTest
Conclusions

J2E Sucks!
Agile Sucks!
DevOps Sucks!
**PERIODIC TABLE OF DEVOPS TOOLS (V2)**

<table>
<thead>
<tr>
<th>Tool</th>
<th>Color</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>GitHub</td>
<td>Yellow</td>
<td>Open Source (OS)</td>
</tr>
<tr>
<td>Gitolite</td>
<td>Red</td>
<td>Freeware (Fr)</td>
</tr>
<tr>
<td>JIRA</td>
<td>Blue</td>
<td>CI (Continuous Integration)</td>
</tr>
<tr>
<td>Bitbucket</td>
<td>Green</td>
<td>Deployment (Dm)</td>
</tr>
<tr>
<td>JIRA</td>
<td>Blue</td>
<td>Cloud / IaaS / PaaS (Pd)</td>
</tr>
<tr>
<td>Bitbucket</td>
<td>Green</td>
<td>BI / Monitoring (En)</td>
</tr>
<tr>
<td>JIRA</td>
<td>Blue</td>
<td>SCM (Source Code Management)</td>
</tr>
<tr>
<td>Bitbucket</td>
<td>Green</td>
<td>Database Mgmt (Os)</td>
</tr>
<tr>
<td>JIRA</td>
<td>Blue</td>
<td>Repo Mgmt (Fr)</td>
</tr>
<tr>
<td>Bitbucket</td>
<td>Green</td>
<td>Config / Provisioning (Dm)</td>
</tr>
<tr>
<td>JIRA</td>
<td>Blue</td>
<td>Release Mgmt (Pd)</td>
</tr>
<tr>
<td>Bitbucket</td>
<td>Green</td>
<td>Build (En)</td>
</tr>
<tr>
<td>JIRA</td>
<td>Blue</td>
<td>Testing (Os)</td>
</tr>
<tr>
<td>Bitbucket</td>
<td>Green</td>
<td>Containerization (Fr)</td>
</tr>
<tr>
<td>JIRA</td>
<td>Blue</td>
<td>Collaboration (Dm)</td>
</tr>
<tr>
<td>Bitbucket</td>
<td>Green</td>
<td>Security (Pd)</td>
</tr>
</tbody>
</table>

**Tools are Bullshit**
Course rationale:
Teaching Devs to speak to Ops

4th year, CS department
Advertise the course as hard

Make it optional
TCF as a reference monolith

Turn-key stack

legacy / iso-functional
@Inject EmailService mailer;
mailer.send(new Email(...));

Devs: Consuming services as calling objects.
@Inject EmailService mailer;
mailer.send(new Email(...));
Ops: Deploying services independently
Ops: Deploying services independently
For us, DevOps doesn’t mean that “one single guy handle both Devs and Ops.”

It means that Devs OR ops can be “on call”
The Dream Team

Guilhem
Philippe
Ben
Seb
Anne-Marie
Industrial Networking (2013-...)

IBM

TREFFTik.
THE CLOUD & JAVA COMPANY

AgilePlayGround.Org

agile tour
Sophia Antipolis 2017

{RIVIERADEV}

Meetup

docker

DEVOPS D-DAY
JEUDI 9 NOVEMBRE 2017
Course Organisation

Full day course (Friday), 55 students max.

Optional, 12 weeks (96h for students)

Project-driven approach (coaching)

Spectrum: From Software Architecture to Ops
Tools are bullshit

We focus on concepts

3rd year: VCS, Tests, Deps
4th Year: Deployment, CI, CD
5th Year: Monitoring & Cloud
### Lectures (16h) + Project Lab (73h) + Eval (7h)

<table>
<thead>
<tr>
<th>Week</th>
<th>Friday Morning</th>
<th>Friday Afternoon</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Introduction to Software Architecture</td>
<td>DevOps Overview</td>
</tr>
<tr>
<td>7</td>
<td>Poly’Event Architecture definition (unsupervised)</td>
<td>Arch. for Testing</td>
</tr>
<tr>
<td>8</td>
<td>Mutation Testing Lab (unsupervised until further notice)</td>
<td>Func. &amp; Int. Tests</td>
</tr>
<tr>
<td>9</td>
<td>Soft. Components</td>
<td>Arch Dojo #1</td>
</tr>
<tr>
<td>10</td>
<td>Poly’Event Project</td>
<td>Poly’Event Project</td>
</tr>
<tr>
<td>11</td>
<td>Winter break</td>
<td>Poly’Event Project</td>
</tr>
<tr>
<td>12</td>
<td>Interoperability &amp; WS</td>
<td>Cont. Integration</td>
</tr>
<tr>
<td>13</td>
<td>Cont. Integration</td>
<td>Build plan &amp; Pipeline</td>
</tr>
<tr>
<td>14</td>
<td>Technical interview (Minimal &amp; Viable Product)</td>
<td>Poly’Event Project (unsupervised)</td>
</tr>
<tr>
<td>15</td>
<td>Persistence</td>
<td>Deployment</td>
</tr>
<tr>
<td>16</td>
<td>Arch Dojo #2</td>
<td>Soft. Containers</td>
</tr>
<tr>
<td>17</td>
<td>Q&amp;A, Stepback</td>
<td>Scaling</td>
</tr>
<tr>
<td>18</td>
<td>Easter Break</td>
<td>Poly’Event Project</td>
</tr>
<tr>
<td>19</td>
<td>Poly’Event Project (unsupervised)</td>
<td>Poly’Event Project</td>
</tr>
<tr>
<td>20</td>
<td>Architecture Exam (3 hours)</td>
<td>DevOps Exam (3 hours)</td>
</tr>
</tbody>
</table>

[https://github.com/mosser/isa-devops](https://github.com/mosser/isa-devops)
Project = Call for bids

2-pages describing a Product Vision

Ski resort management

The Cookie Factory

Un-loyalty program

Messaging System