

01 - Introduction

Multi-tiered architecture, Java EE

AMT 2020
Olivier Liechti

Introduction to multi-tiered architectures & Java EE

↪ Dotnet

~~Java~~ pas le sujet principal

Break

Getting to know the application servers (live demo)

Outils à disposition

MVC, Pipes and Filters and IoC patterns

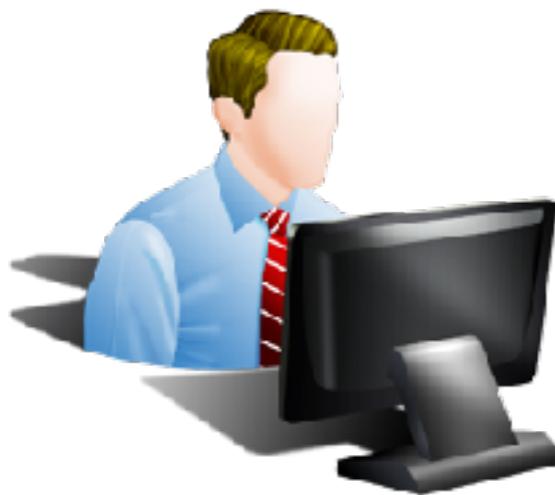


Introduction

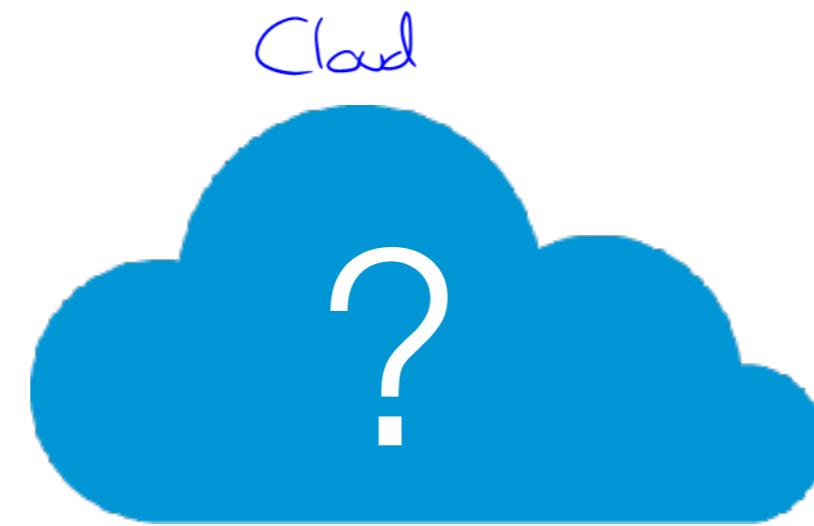
Multi-Tiered Applications



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch



Utilisateur avec
un device connecté
au réseau



Reçoit des cmd
de l'utilisateur



Données

Multi-Tiered Applications



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch



The application is **structured** in “tiers” (tranches)

Chaque boîte est une API

Multi-Tiered Applications

Frameworks



EHCA~~CHE~~



APIs



- Client
- servlet
- JSP
- JAX-RS
- JAX-WS



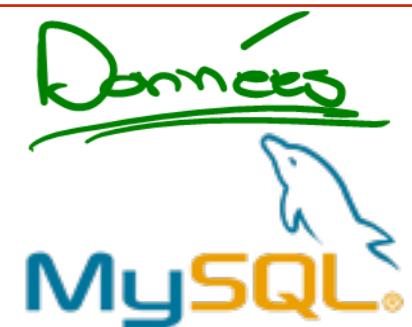
Java EE
Application Server

EJB

JDBC

JPA

JMS



Tools

Outils de test



Automatisation



~~Jenkins~~

Github Actions





HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

MULTI-TIERED APPLICATION DEVELOPMENT: FROM LEGACY TO MODERN JAVA EE

OLIVIER LIECHTI

A hands-on approach to design patterns and engineering practices

2018 – v0.4

Bonjour,

J espère que vous allez bien.

Aujourd'hui j ai eu la chance de rencontrer Ken Duda qui est le CTO et un des fondateurs de arista networks. Le discours qui l'a tenu m'a tellement rappelé votre cours d'amt que je me suis senti obligé de vous écrire ce message. Comme quoi de la heig à une multinationale côté au Nasdaq en passant par les starts up ce n est pas la quantité ni la rapidité qui prime mais la qualité. Ce qui implique des tests automatisés de qualités (peut être pas avec cucumber il faut pas exagérer 😂).

Si vous avez 20 min à perdre je joins un lien où il délivre à peu près le même message (vous allez y retrouver votre message sur les tests 😂)

[https://www.google.ch/url?
sa=t&source=web&rct=j&url=https://m.youtube.com/watch
%3Fv
%3D
%3DVdjZq4dRjf4&ved=2ahUKEwjayOzguvzdAhUk_SoKHZ-xCzsQt9IBMAV6BAgJECM&usg=AOvVaw2_nqONCTi58MIGzRHTXFS](https://www.google.ch/url?sa=t&source=web&rct=j&url=https://m.youtube.com/watch%3Fv%DVDjZq4dRjf4&ved=2ahUKEwjayOzguvzdAhUk_SoKHZ-xCzsQt9IBMAV6BAgJECM&usg=AOvVaw2_nqONCTi58MIGzRHTXFS)

Bonne soirée et j espère vous croiser à noël pour l apero de tweb et amt



Si je peux me permettre un commentaire sur les cours, que vous pourrez peut-être relayer plus tard. C'est qu'on voit trop peu de JavaEE, en comparaison aux autres cours et aux offres d'emploi du marché. Ça fait une semaine que je regarde tous les jours les offres et je pense que le 80% des offres c'est Java/JavaEE. Et personnellement, même si j'avais bien compris en AMT, le fait qu'on en ait vu si peu, me donne l'impression de ne pas être assez prête pour être engagée pour ça et c'est assez embêtant. Même si les concepts de base sont quand même acquis

August 18

Bonjour,

J'ai eu un entretien avec [REDACTED] (une boîte de 150 personnes). C'est normal que pour l'entretien technique, ils demandent de faire une API documentée avec Swagger (exactement ce que vous nous avez demandé de faire pour la deuxième partie du semestre en AMT, merci  ) et que je le fasse en quelques heures?

edited 17:12



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

August 30

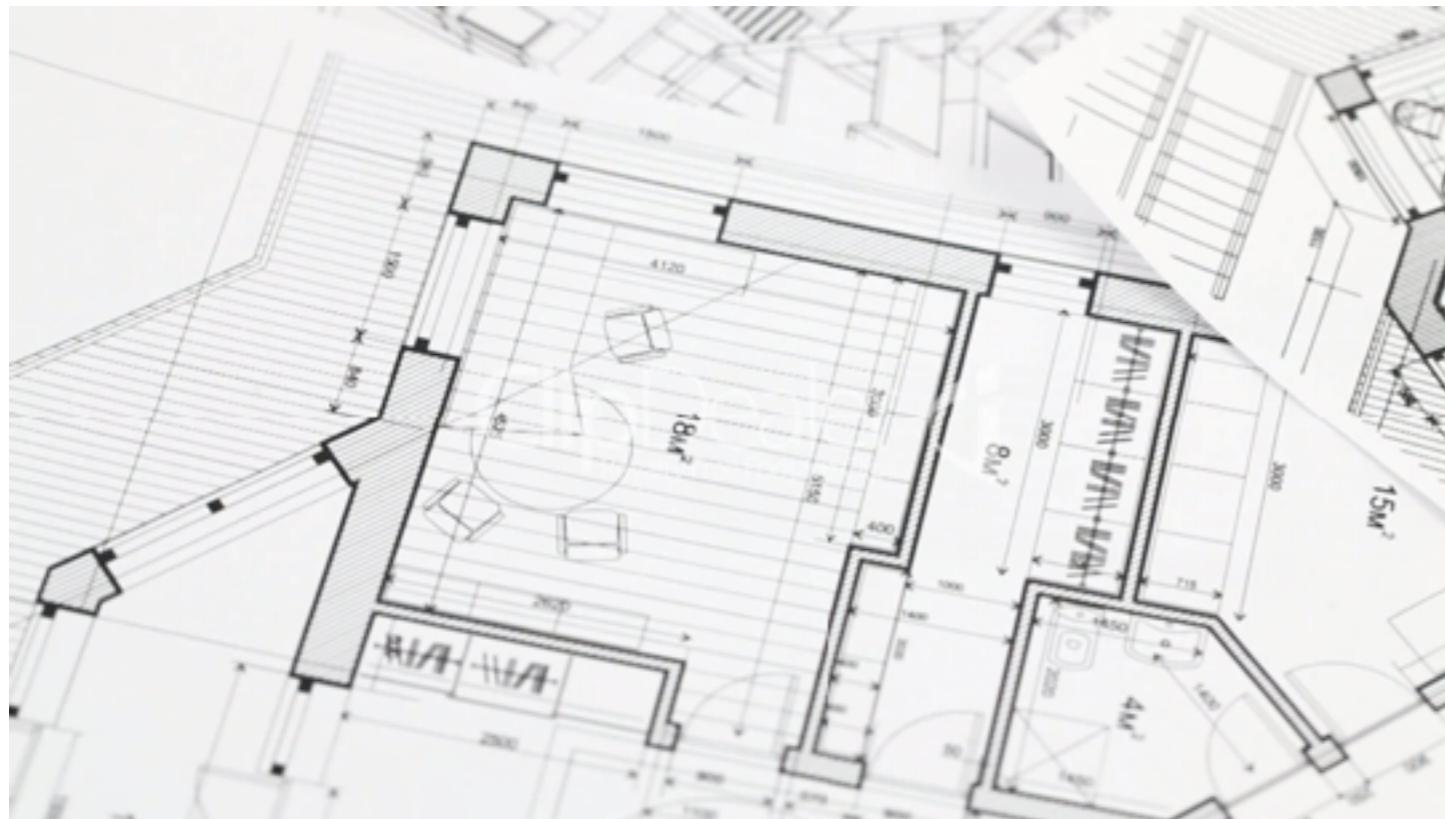
Bonjour, comment allez-vous ? J'espère que ça va bien.

Il y a quelque jours, je repensais à la HEIG-VD et aux cours que j'avais eu là bas. Je me souviens de vos cours, surtout AMT où on apprenait à utiliser des APIs, et je m'en rends compte que ça m'a bien aidé dans mon travail de tous les jours.

Le fait d'avoir suivi vos cours et labos m'ont permis de comprendre plus facilement certains concepts de programmation API et réaliser certaines tâches/projets en étant beaucoup plus à l'aise que si je n'avais pas eu le cours.

J'imagine qu'en tant qu'enseignant vous avez pas dû avoir beaucoup de feedback de vos élèves après leur première expérience professionnelle et je voulais donc vous apporter le mien. Merci pour la qualité de vos cours.

edited 15:17



Architecture

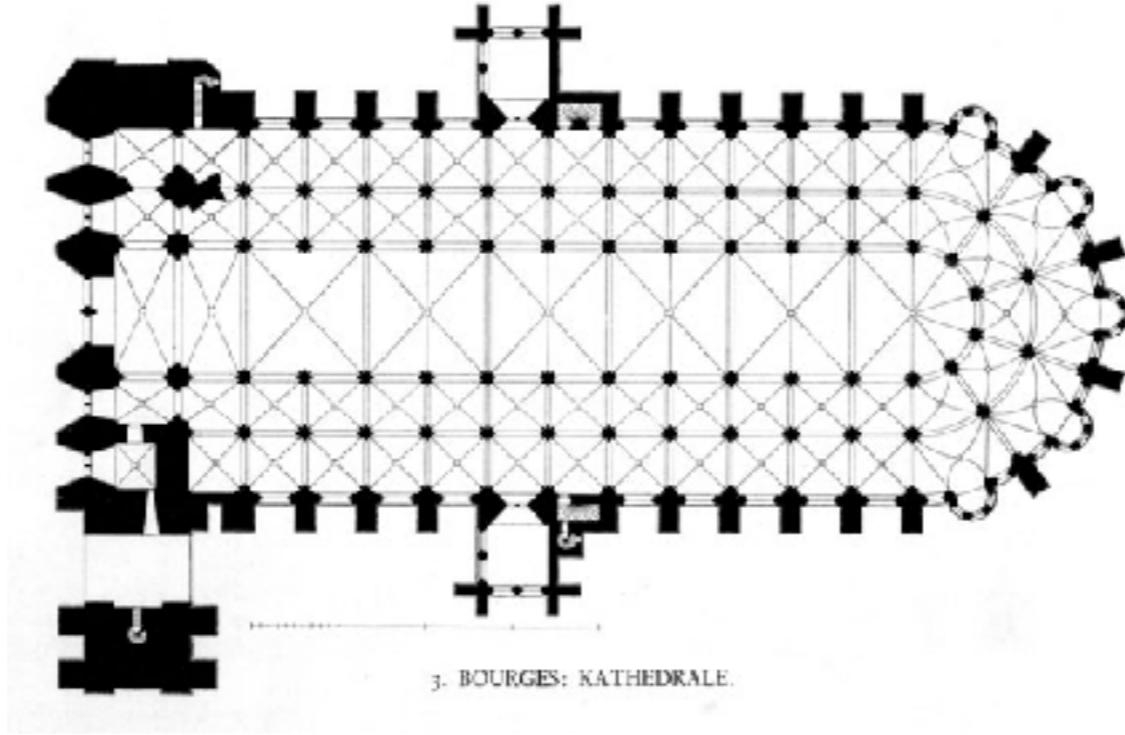
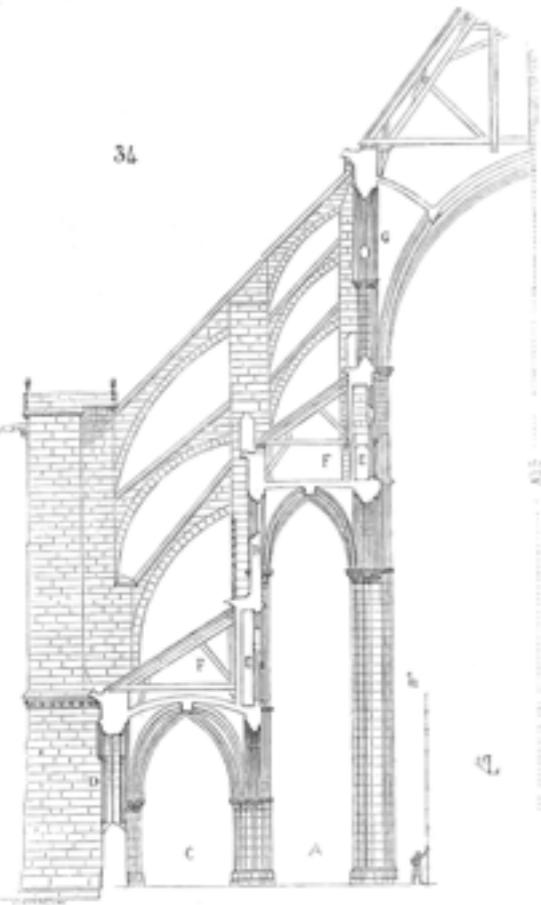


Java EE ↑ dans abstraction
=> comprendre la base
et simplifier la tâche



What do we mean by “**architecture**” “**architectural style**” in the context of software systems?

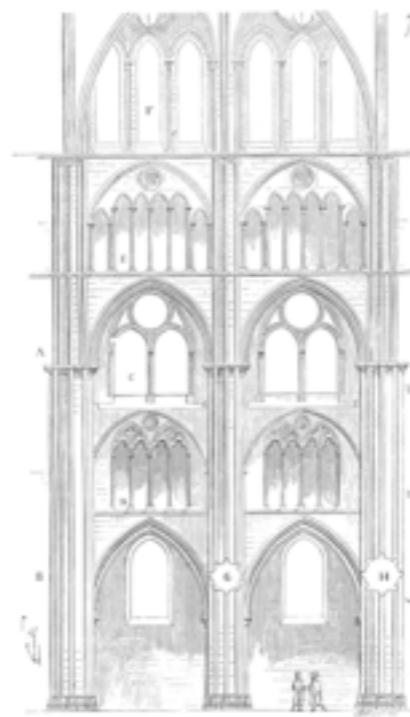
- There is **no single definition** for software architecture!
- The **architecture of a particular application** is a **description** of its **high-level structure** and **behavior**. It is a description of its **components**, how they **interact** with each other and with **external systems**.
- The **architecture of one particular application** is **documented** with a **collection of models** (**UML** diagrams are examples of such models).
- An **architectural style** is a **set of core principles and rules** (**architectural patterns**), which are **common** to a large number of **different applications**.

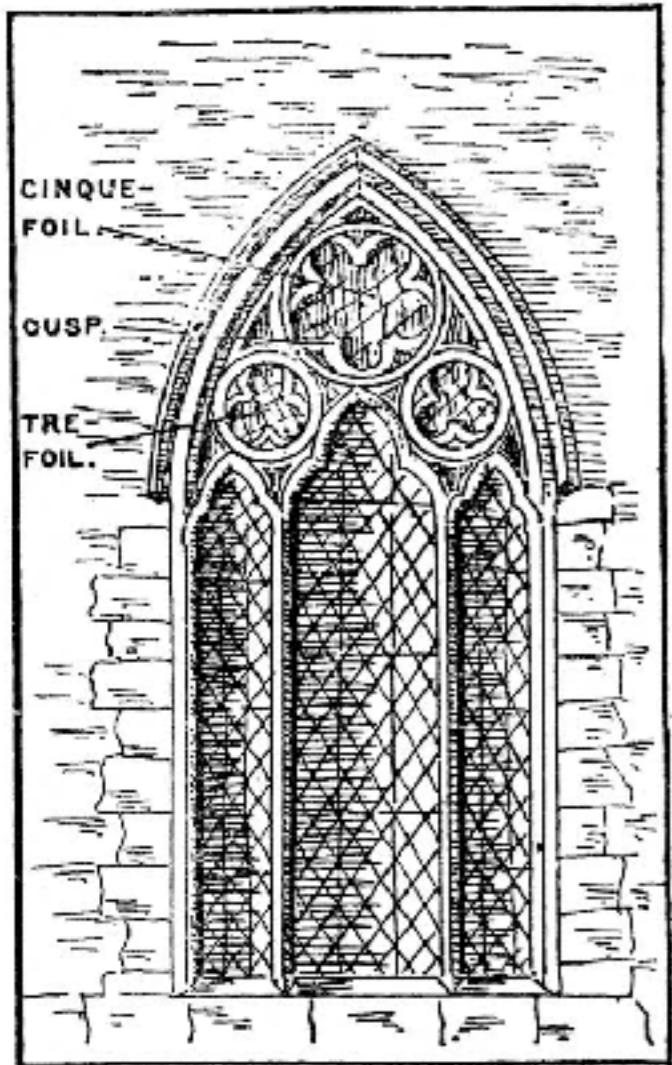
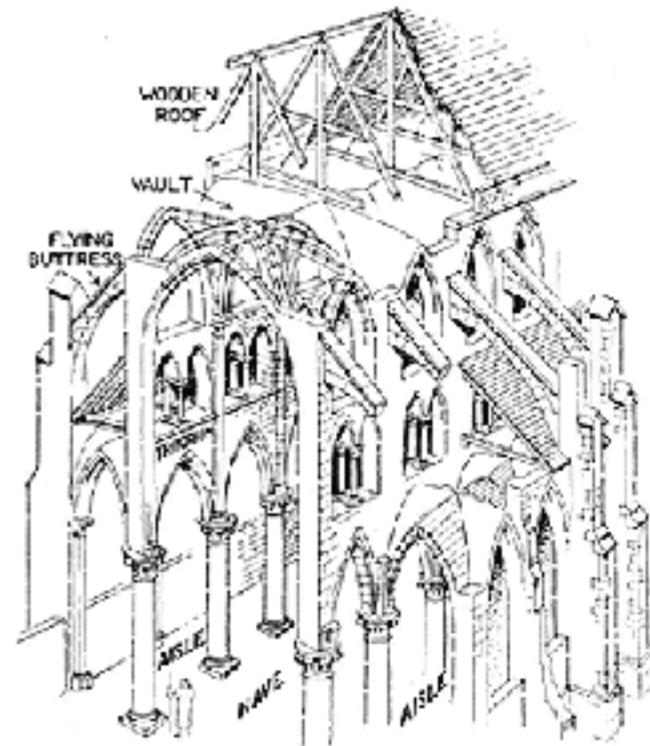
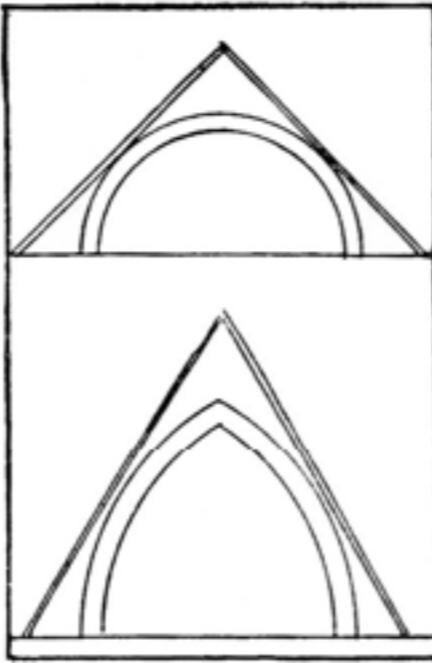
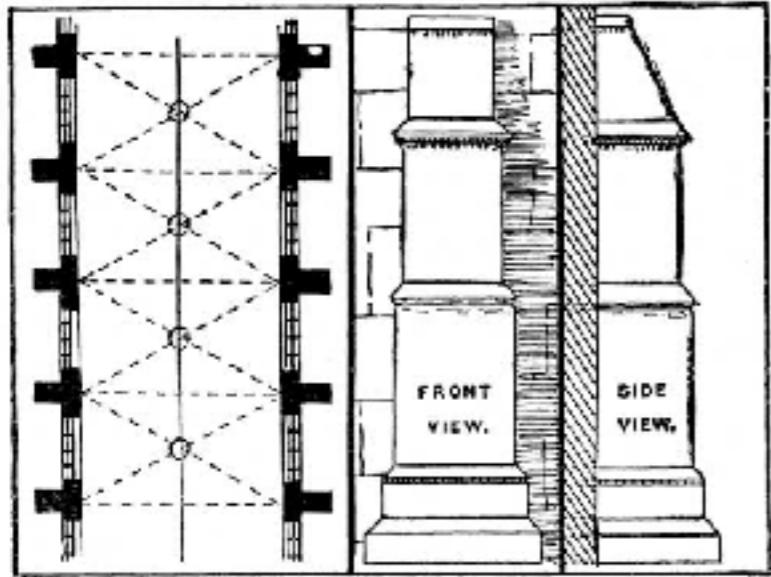


The Architecture of Bourges Cathedral

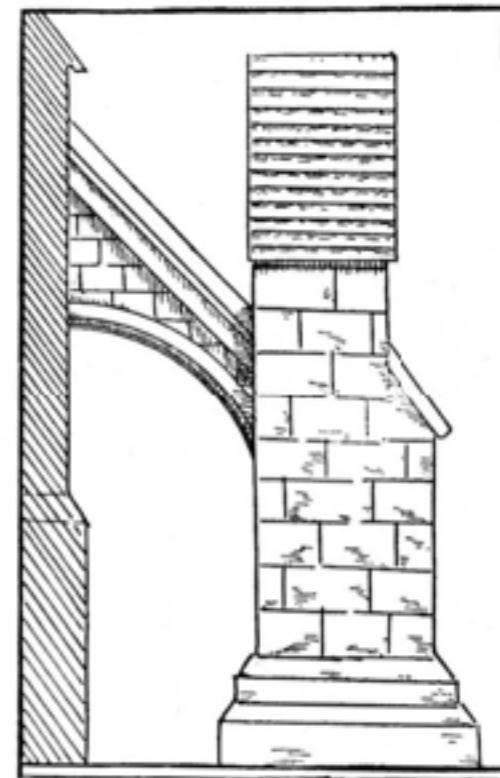


HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch





The Gothic Architectural Style





What are some of the well-known **architectural styles?**

- The client-server architectural style
- The 3-tiered architectural style
- **The multi-tiered architectural style**
- The RESTful architectural style
- The service oriented architectural style
- The event-driven architectural style
- The layered architectural style

 Architectural styles are **not mutually exclusive**. Many applications combine patterns from several styles!

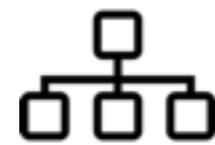


3-tiers, multi-tiers... **how many tiers** does a multi-tiered application have, and which ones?

But: isoler les choses, φ interdépendances



human-readable
markup generation



Page navigation



Data access



"Règles de conduite"

Shared business
logic and **rules**



φ dépendances



machine-readable
markup generation

Extradition des requêtes
HTTP et les transférer



Message/
event bus

Bus de messages



DDD, file system

User

Client

Presentation

Business
≈ domain

Integration
≈ data access

Resources



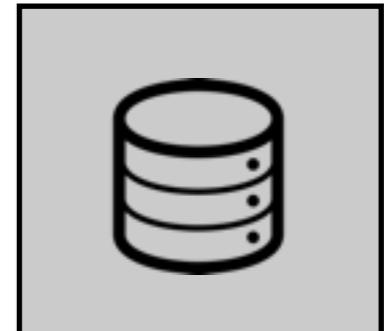
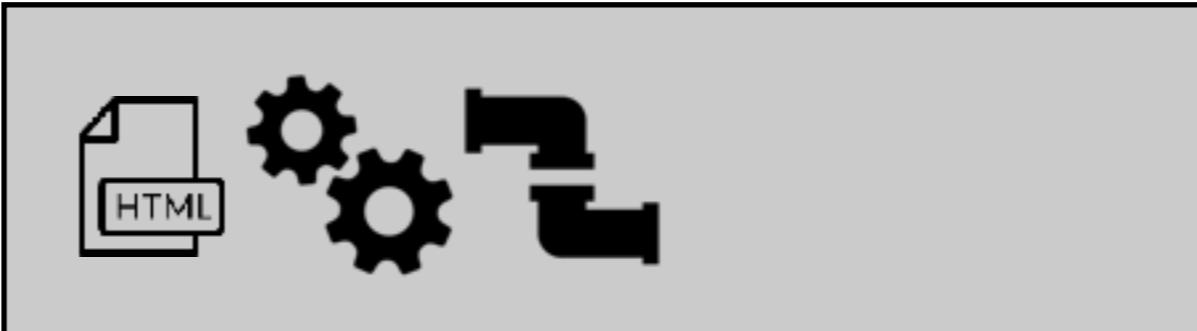
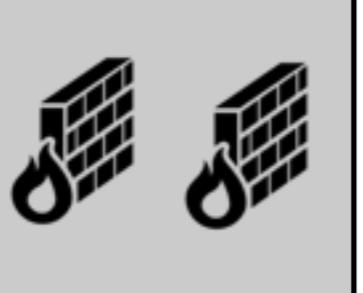
What is the mapping between tiers, **logical nodes** and **physical nodes**?



Dev environment



Minimal environment



Simple environment

User

Client

Presentation

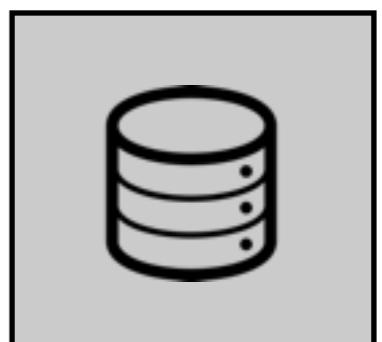
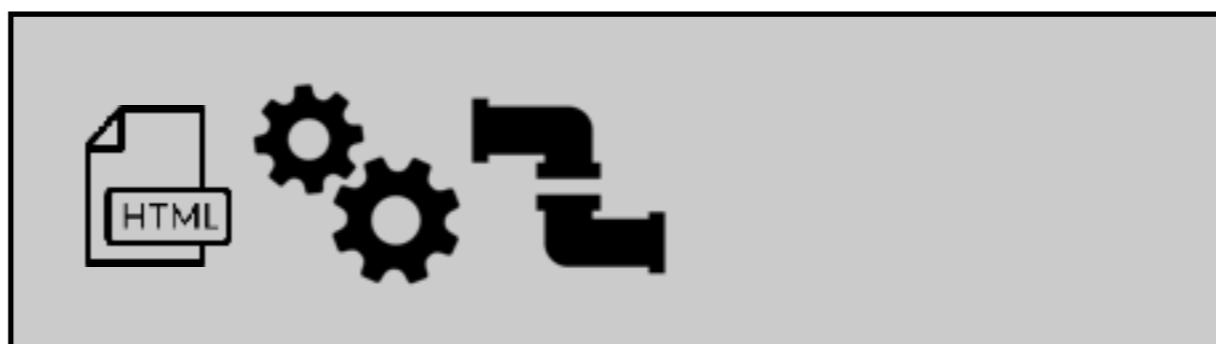
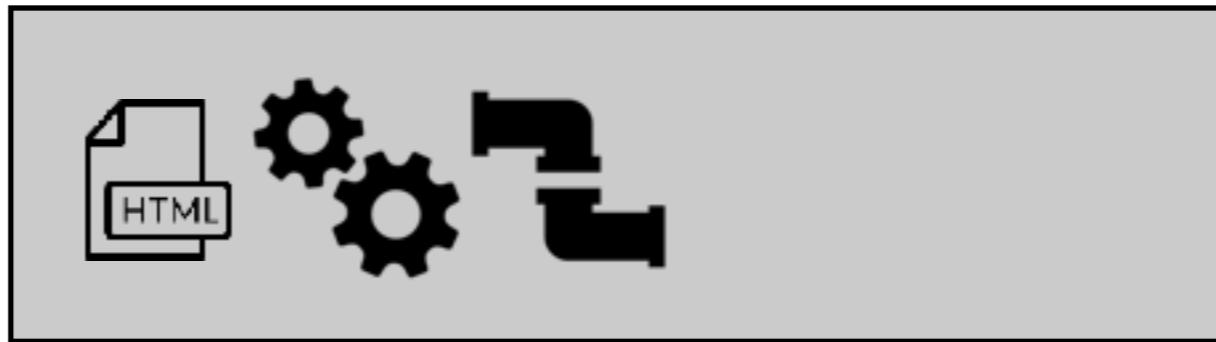
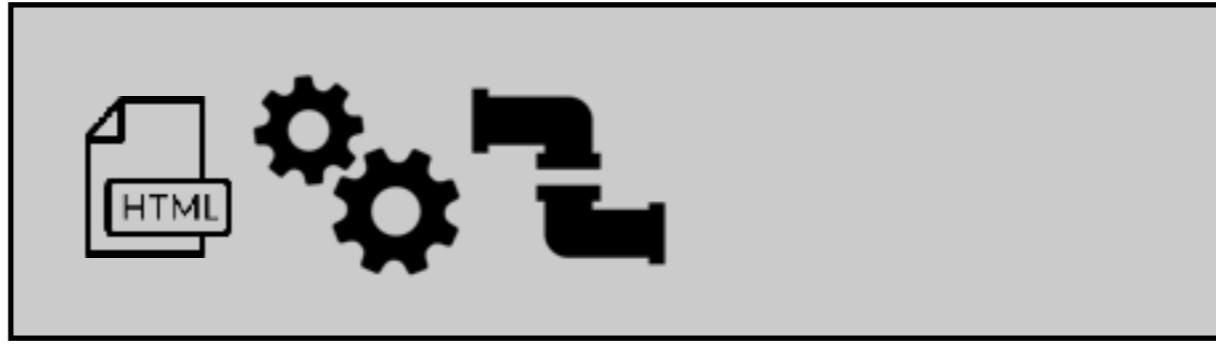
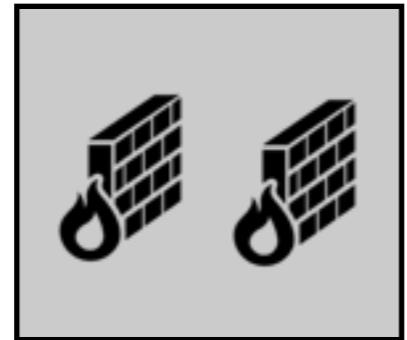
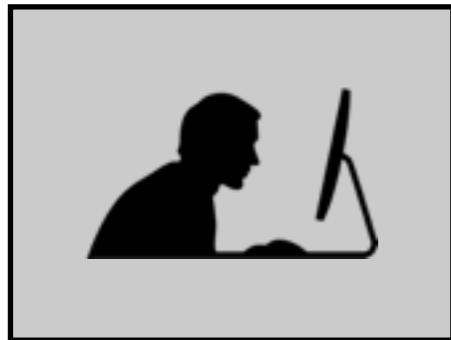
Business

Integration

Resources



What is the mapping between tiers, **logical nodes** and **physical nodes**?



Typical environment for scalability and availability

User

Client

Presentation

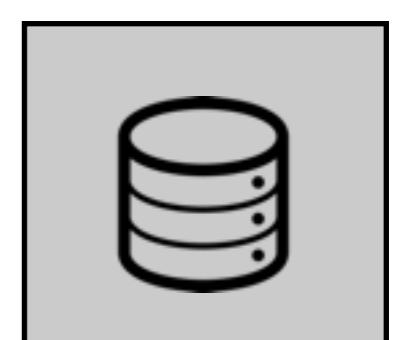
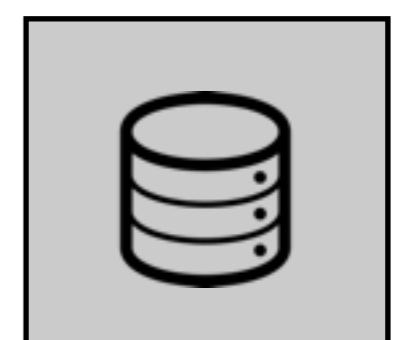
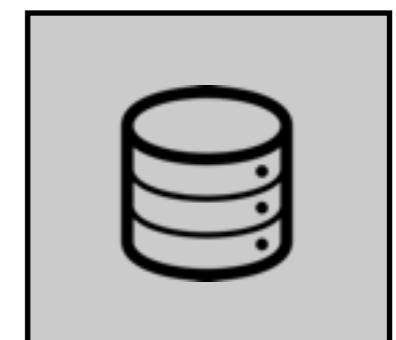
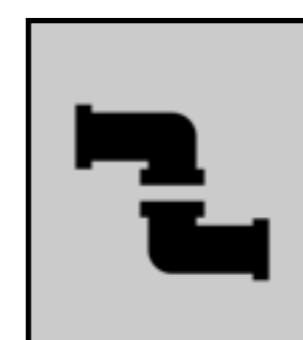
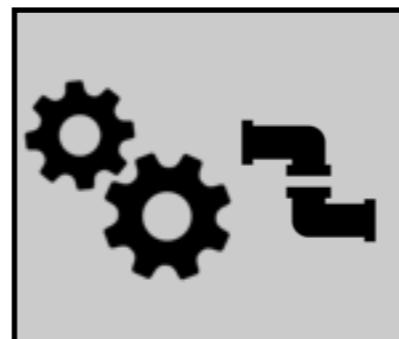
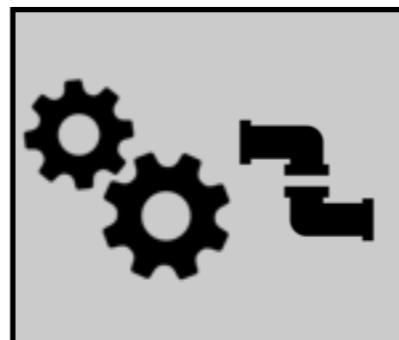
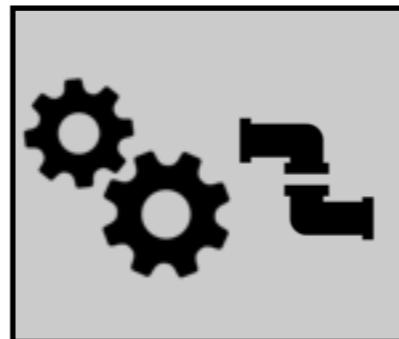
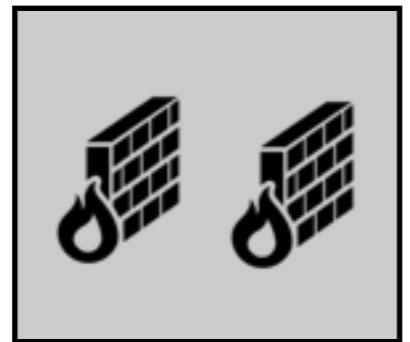
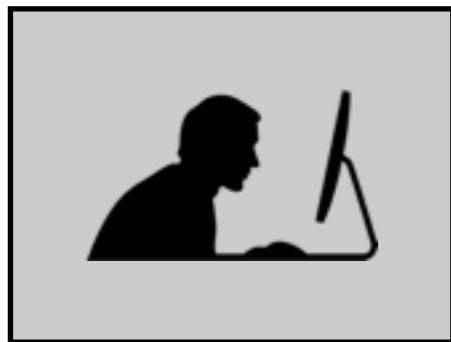
Business

Integration

Resources



What is the mapping between tiers, **logical nodes** and **physical nodes**?



Fully distributed environment

User

Client

Presentation

Business

Integration

Resources



What is the mapping between tiers, **logical nodes** and **physical nodes**?

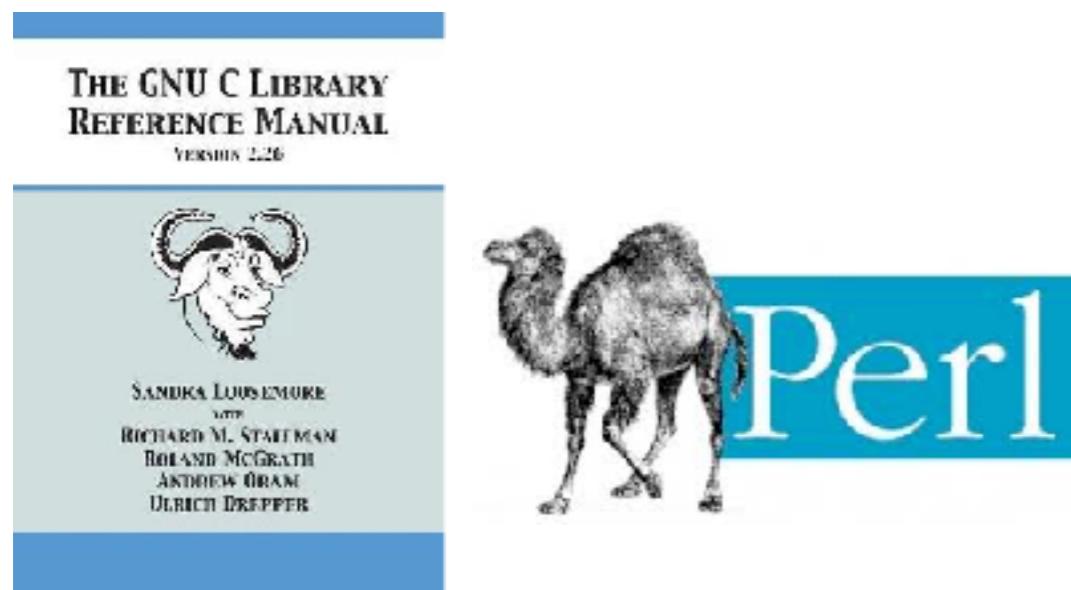
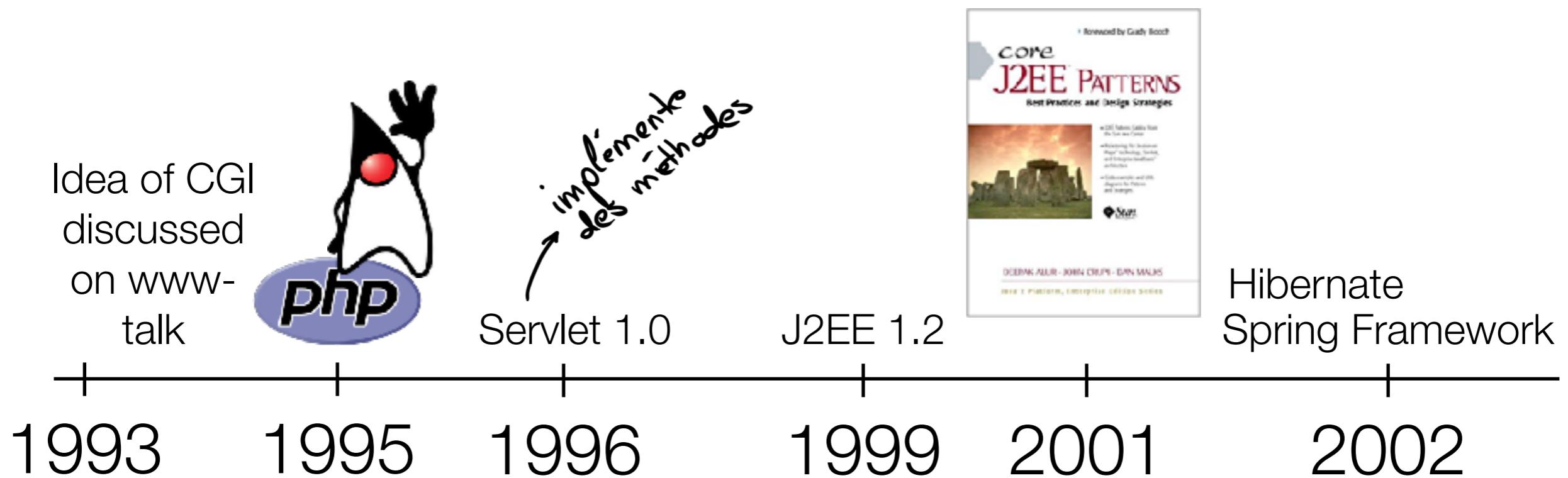
- Just because it is **possible to physically separate the presentation and the business tier** does not mean that it is always a good idea!
- One reason to keep them in the same node is that **invoking a remote service is a lot slower than invoking a local service**.
- Introduce a **reverse proxy** as soon as possible. In simple environments (e.g. dev.), it does not require a physical node.
- **Database nodes** are often shared by several applications. Optimization techniques are fairly different for database and application nodes. They are often physically separated.

(Rapidly evolving) virtualization technologies gives us more options. Think about **Docker**, which allows you to “run a data center” on your laptop.



Java Enterprise Edition

Emergence of web development platforms



What is Java Enterprise Edition?

- It is a **development platform**: it provides high-level APIs to develop software components.
- It is an **execution platform**: it provides an environment to deploy and bring these components “to live”.
- It is an **enterprise platform**: it provides support for distributed transactions, security, integration, etc.
- **Separation of concerns**: "The developer takes care of the business logic. Java EE takes care of the systemic qualities".



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch



http://flickr.com/photos/decade_null/427124229/sizes/m/#cc_license

Java EE and standards



- **Java EE is a specification**
 - Defined through the JCP, it is a specification that software editors can decide to implement. Java EE 5 is defined in JSR 244.
- **Java EE is an “umbrella” specification**
 - Java EE builds upon other specifications (servlets, EJBs, JDBC, etc.) and specifies which specifications (and which versions) need to be implemented by a Java EE certified application server.
 - Java EE also defines a programming model and defines several roles (developer, assembler, deployer, etc.).

Specification Lead
 Bill Shannon Sun Microsystems, Inc.

Expert Group
Barreto, Charlton
Capgemini
Dudney, Bill
Hewlett-Packard
Kohen, Elika S.
Oracle
Pralap, Rama Murthy Amar
Rehnshagen, Dirk
Shah, Suneet
Tiwari, Ashish
Umapalay, Sivasundaram

BEA Systems
Chandrasekaran, Muralidharan
E.piphany, Inc.
IBM
Leme, Felipe
OW2
Raible, Matt
SAP AG
Sun Microsystems, Inc.
Tmax Soft, Inc.

Borland Software Corporation
Crawford, Scott
Genender, Jeff
Ironflare AB
Novell, Inc.
Pramati Technologies
Red Hat Middleware LLC
SeeBeyond Technology Corp.
Sybase
Trifork



J2EE or Java EE?



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

- Java Enterprise Edition is **not** a recent specification.
- The SDK 1.2 was published in 1999.
- The specification is managed through the JCP since version 1.3
- We used to talk about "Java 2 Enterprise Edition 1.3", or J2EE 1.3
- We then moved from J2EE 1.3 to J2EE 1.4 to... **Java EE 5**
- Today, we should speak of Java EE, but J2EE is still sometimes used...
- Java EE 7 was released on May 28th, 2013
- Java EE 8 was released on August 31st, 2017. Seriously?



Watch out! Java EE (formerly known as J2EE)
has evolved a lot over the years!

I personally thought that Java EE, as an umbrella specification, was dead.

The JSRs that are part of Java EE (servlets, JAX-RS) would continue to live and evolve, but they will be used directly by independent application frameworks (like Spring)



Then, this happened, in 2018...



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

The screenshot shows the Jakarta EE website homepage. At the top right, there are links for "Edit my account" and "Manage Cookies". Below that is a navigation bar with links for "About", "Members", "Connect", "More...", and a search icon. The main title "Jakarta EE" is prominently displayed in the center. Below the title, the tagline "The New Home of Cloud Native Java" is shown. A sub-tagline below it reads: "Powered by participation, Jakarta EE is focused on enabling community-driven collaboration and open innovation for the cloud." At the bottom of the main section are two orange buttons: "Jakarta EE Working Group" and "Stay Connected".

Strategic Members



ORACLE



Tomitribe

Participating Members

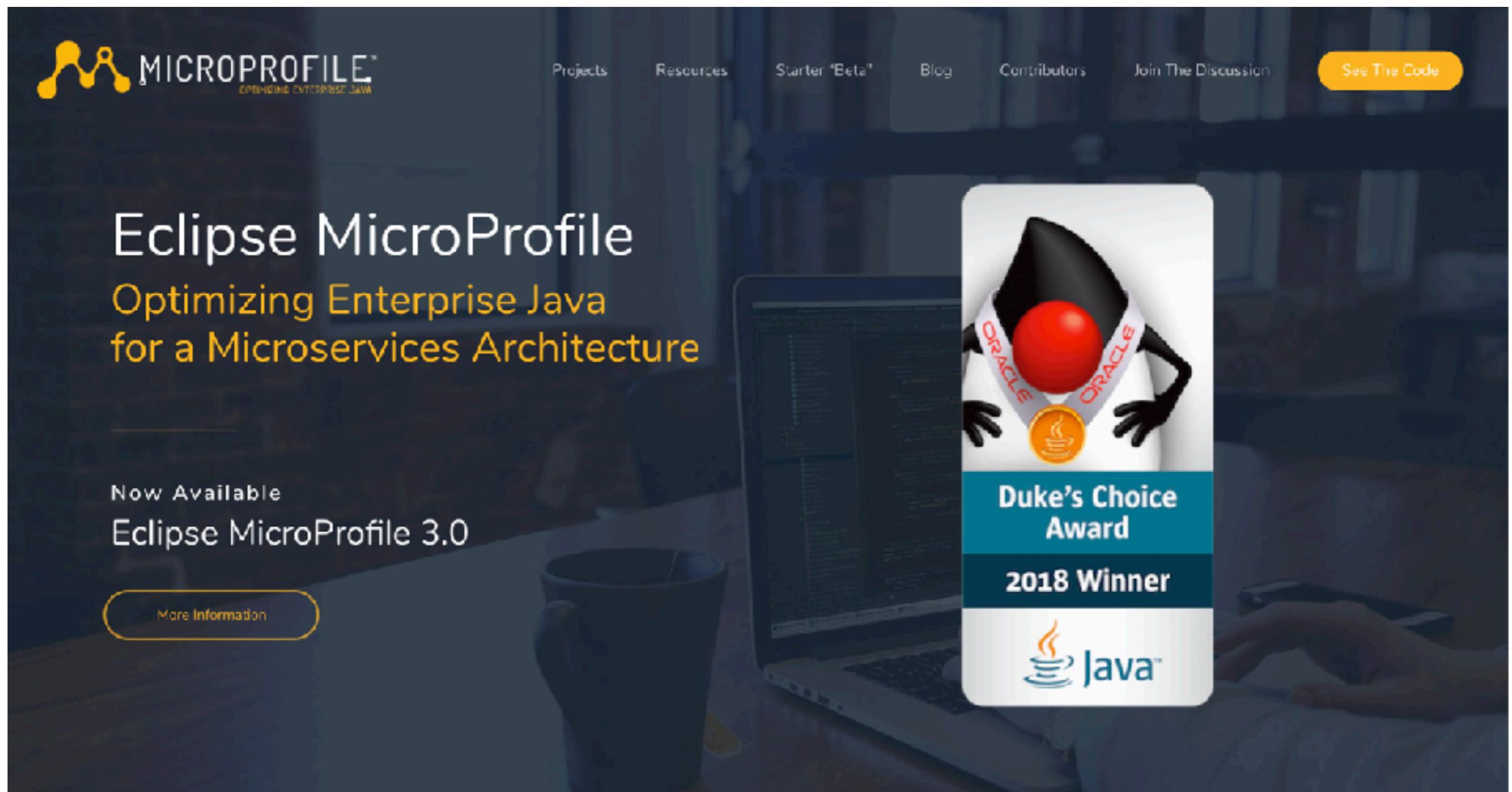


[webtide](#)

and this, ...



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch



The banner features the Eclipse MicroProfile logo and slogan "OPTIMIZING ENTERPRISE JAVA". It highlights the availability of Eclipse MicroProfile 3.0 and displays the "Duke's Choice Award 2018 Winner" badge, which includes the Java logo.

MICROPROFILE[™]
OPTIMIZING ENTERPRISE JAVA

Projects Resources Starter "Beta" Blog Contributors Join The Discussion See The Code

Eclipse MicroProfile

Optimizing Enterprise Java for a Microservices Architecture

Now Available
Eclipse MicroProfile 3.0

More Information

Duke's Choice
Award
2018 Winner

Java



Tomitribe

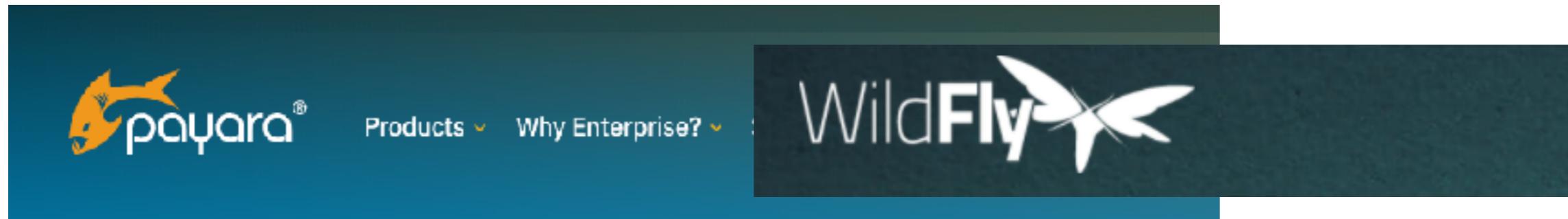


GlassFish

The Open Source Java EE Reference Implementation



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch



AMT 2020

Un important
pour le projet

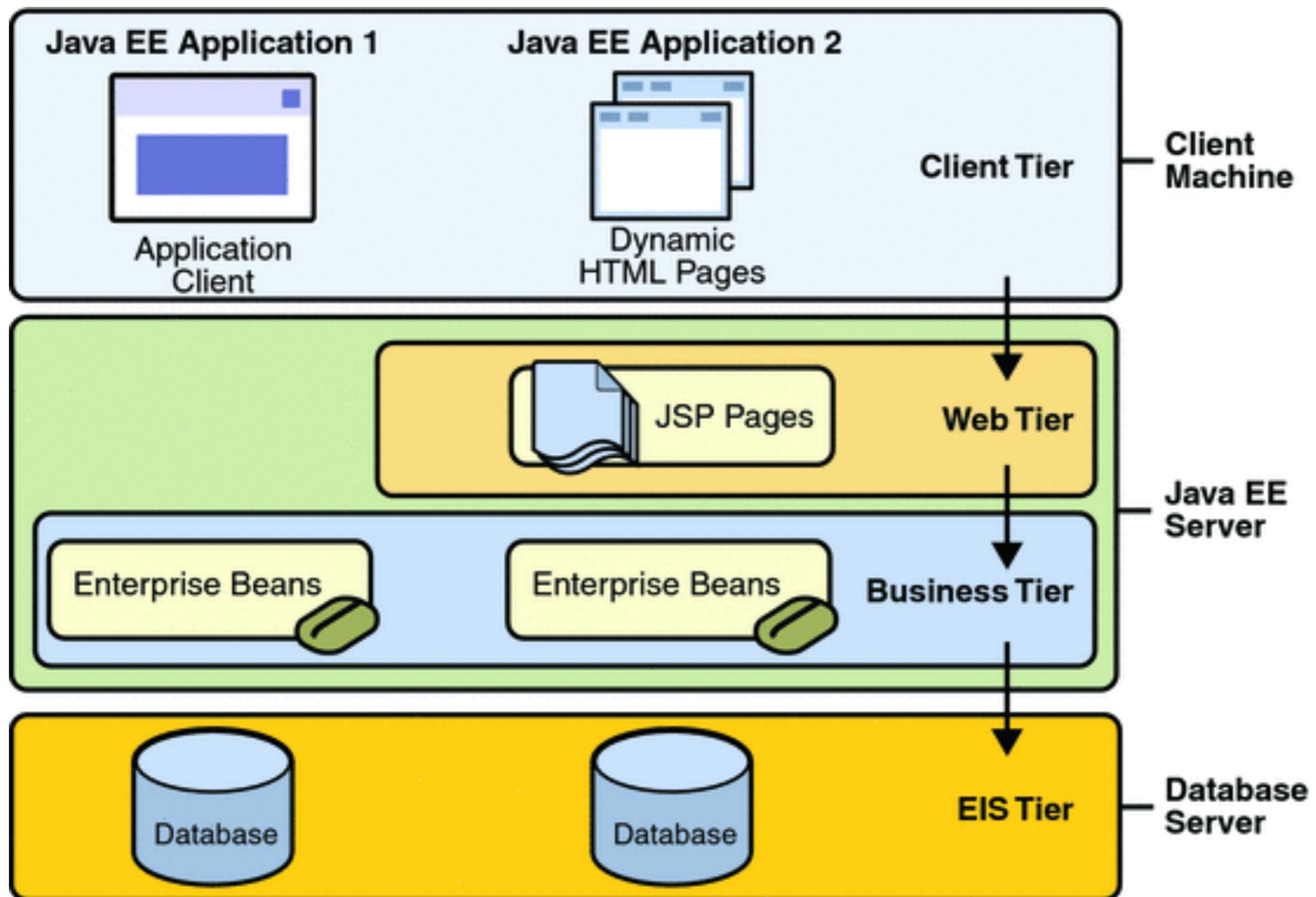


The screenshot shows the Open Liberty website homepage. At the top, there's a navigation bar with links for "Get Started", "Guides", "Docs", "Support", and "Blog". The main feature is a large "Open Liberty™" logo with the subtitle "An IBM Open Source Project". Below the logo, a tagline reads "A lightweight open framework for building fast and efficient cloud-native Java microservices." A descriptive paragraph follows, stating "Build cloud-native apps and microservices while running only what you need. Open Liberty™ is the most flexible server runtime available to Java™ developers in this solar system." At the bottom of the main content area is a green button labeled "Get Open Liberty". To the right of the main content, there's a cartoon illustration of a person sitting on a green planet, wearing a blue space helmet and holding a tablet, with a beam of light from the helmet illuminating the planet. The planet has small green plants growing on it.

Architecture



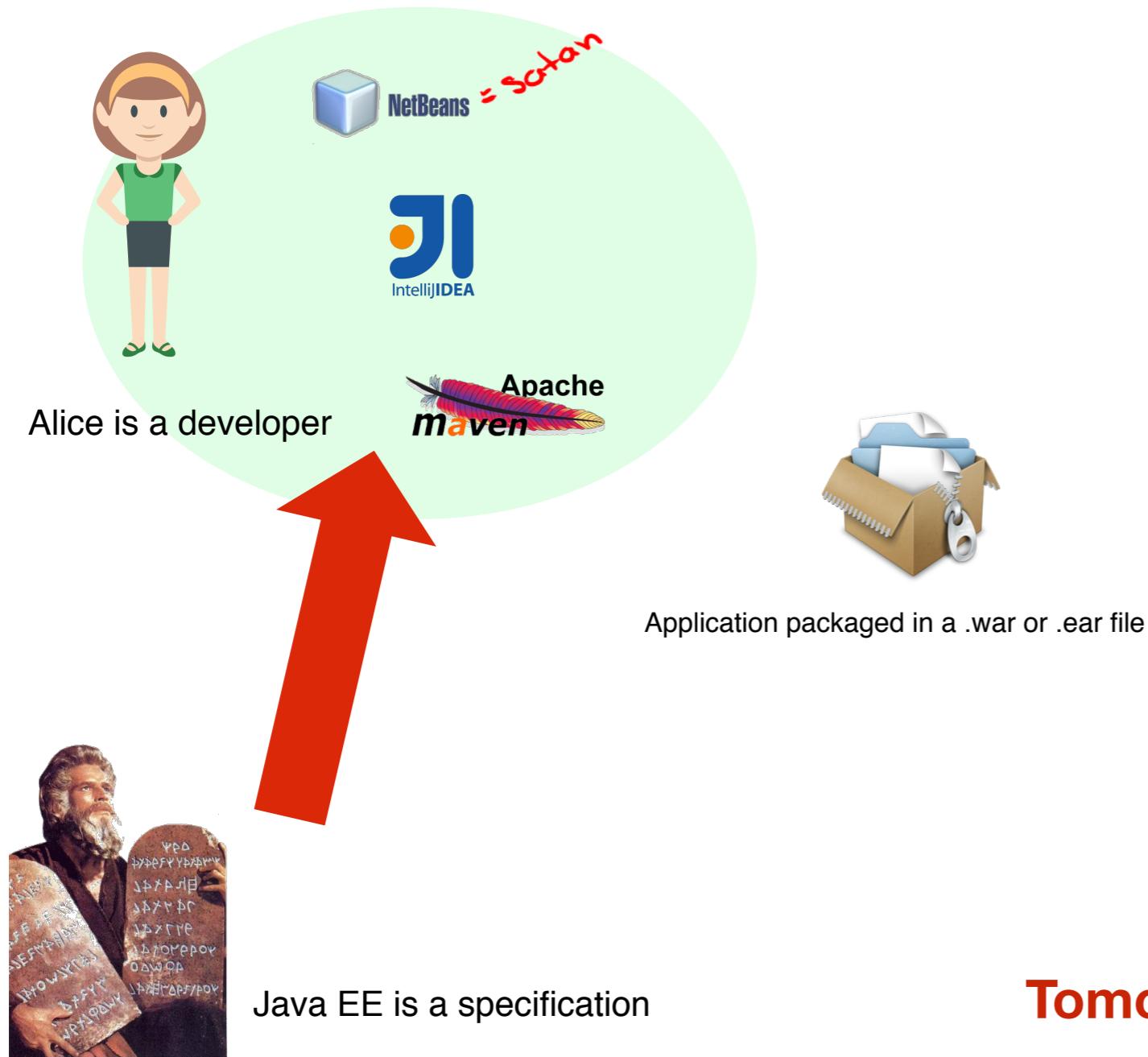
- The software that implements the Java EE specification is called an “**application server**”
 - There are **open source** and **proprietary** application servers.
 - Glassfish, JBoss, WebSphere, BEA WebLogic are examples of application servers.
 - Editors compete on aspects that are not defined the specification (clustering, administration, etc.).
- Key notion in the Java EE architecture: the **containers**
 - a container is an **environment** in which we deploy components;
 - a container **provides services** (transactions, security, etc.) through APIs;
 - there are different containers in Java EE: the "**web**" container, the "**ejb**" container and even a "**client**" container that can be used for rich clients.



AMT-Discovery



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch



AMT-Discovery



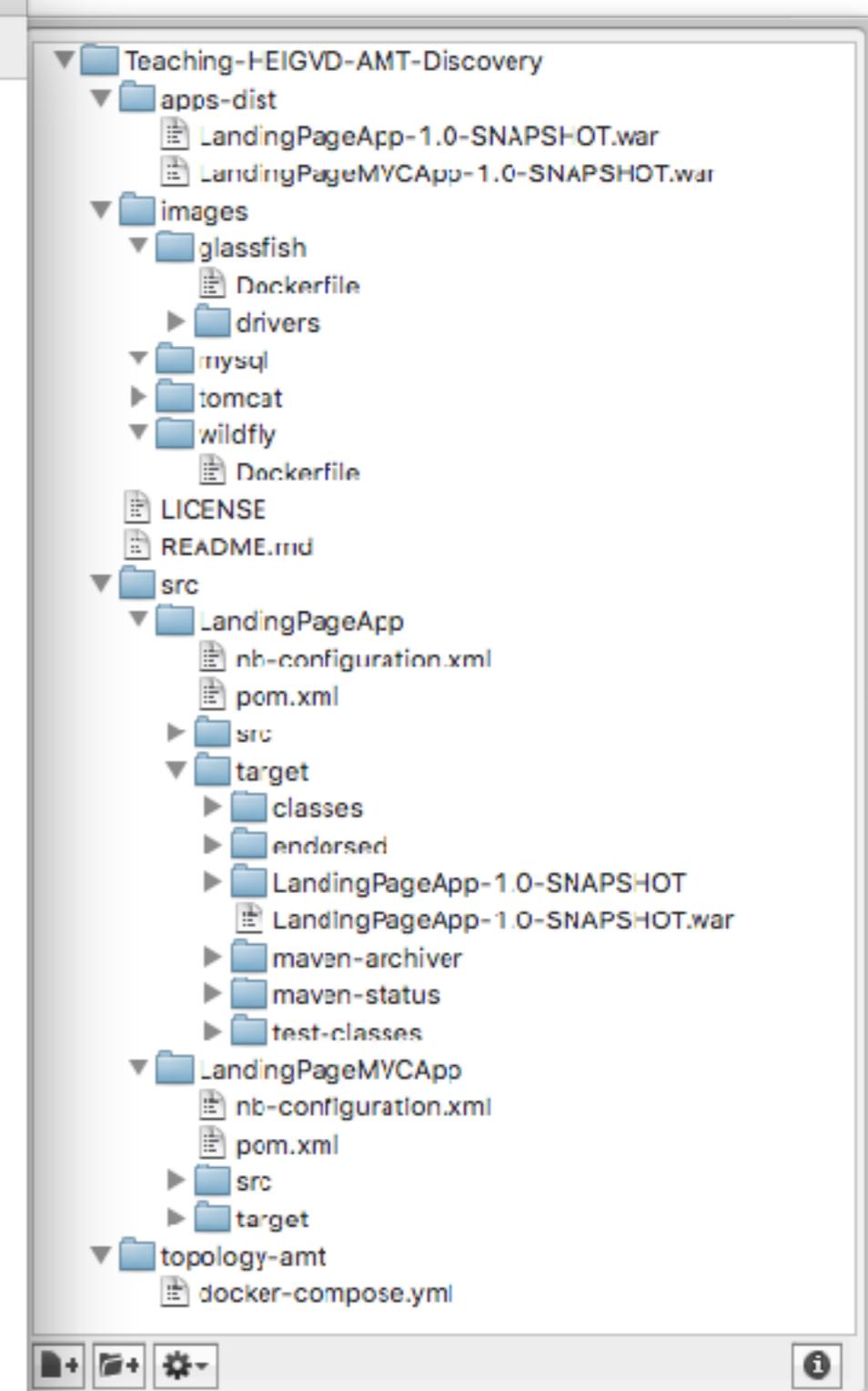
HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

```
* docker-compose.yml — Teaching-HEIGVD-AMT-Discovery
  * docker-compose.yml

1  version: '2'
2  services:
3    glassfish:
4      build: ./images/glassfish
5      ports:
6        - "8080:8080"
7        - "4848:4848"
8    wildfly:
9      build: ./images/wildfly
10     ports:
11       - "9090:8080"
12       - "9990:9990"
13    tomcat:
14      build: ./images/tomcat
15      ports:
16        - "7070:8080"

  voir docker
```

Line: 16 Column: 19 YAML Soft Tabs: 4



AMT-Discovery



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

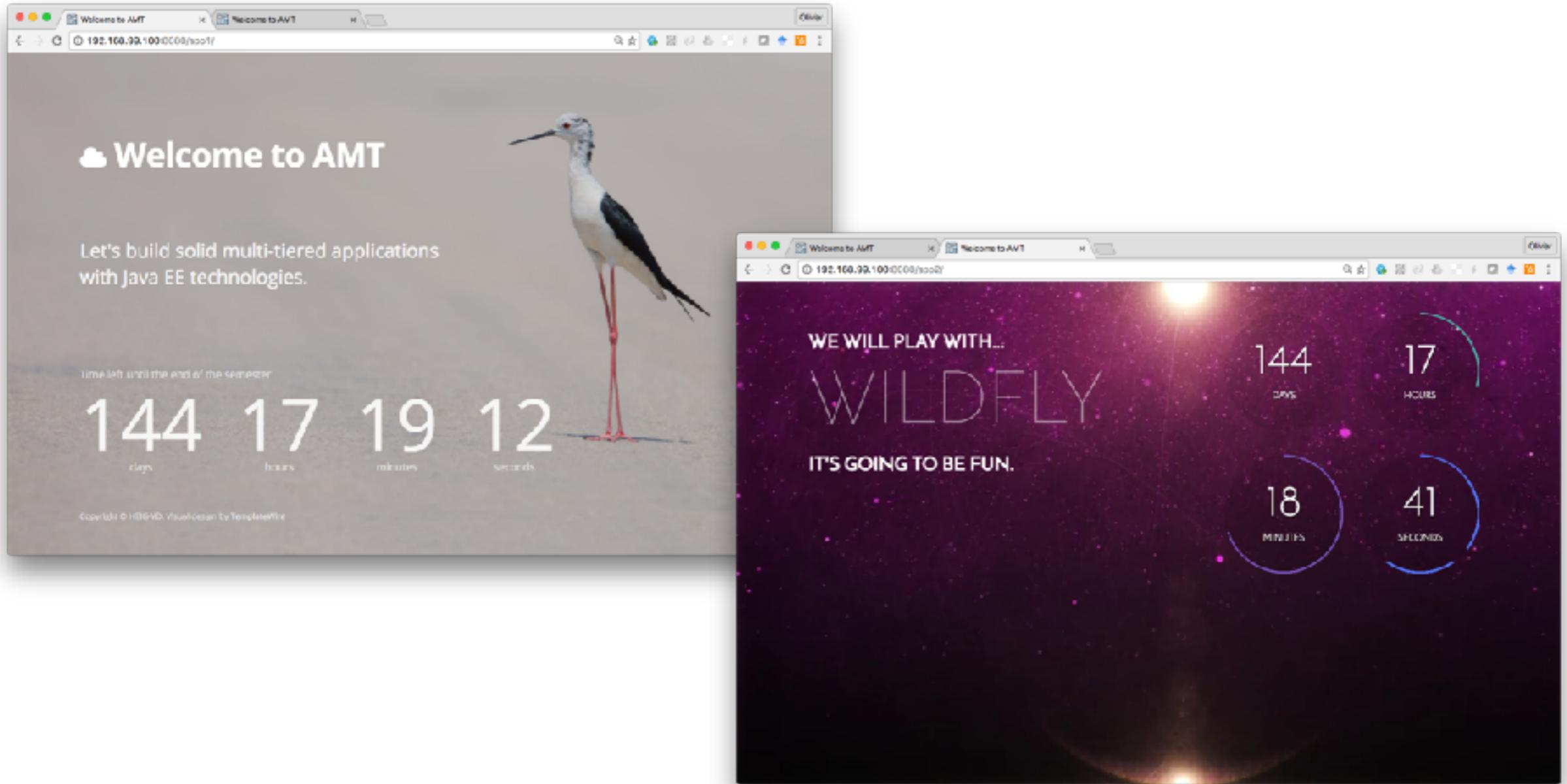
The screenshot displays three separate browser windows side-by-side:

- GlassFish Console - Common**: Shows the GlassFish Server Open Source Edition interface with a sidebar for managing Tomcat, including sections for Managing Tomcat, Release Notes, and Migration Guide.
- Apache Tomcat/8.5.5**: Shows the Apache Tomcat 8.5.5 interface with a message indicating successful installation and a developer quick start section.
- WildFly Management**: Shows the WildFly 8.5.5 management interface, which is a fork of GlassFish. It includes sections for Deployments, Configuration, and Runtime.

AMT-Discovery



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch





What do we want learn?
How to drive a car?
How a car works?

What makes it easier to learn about

How to drive a car?

How a car works?



Questions?



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch