

Jakarta EE

MicroProfile

architectures

adam-bien.com

"It's not work if you like it"
...so I never worked. #java



adam-bien.com

adambien.blog

airhacks.io

airhacks.tv

airhacks.news

airhacks.fm

airhacks.live

adam-bien.com

airhacks.TV

adam-bien.com

workshops.adam-bien.com

NEW Virtual Online Workshops, Winter 2020:

MicroProfile with Quarkus, December 10th, 2020

**Micro Frontends with Web Components, lit-html and redux,
December 11th, 2020**

[past workshops / catalog](#)

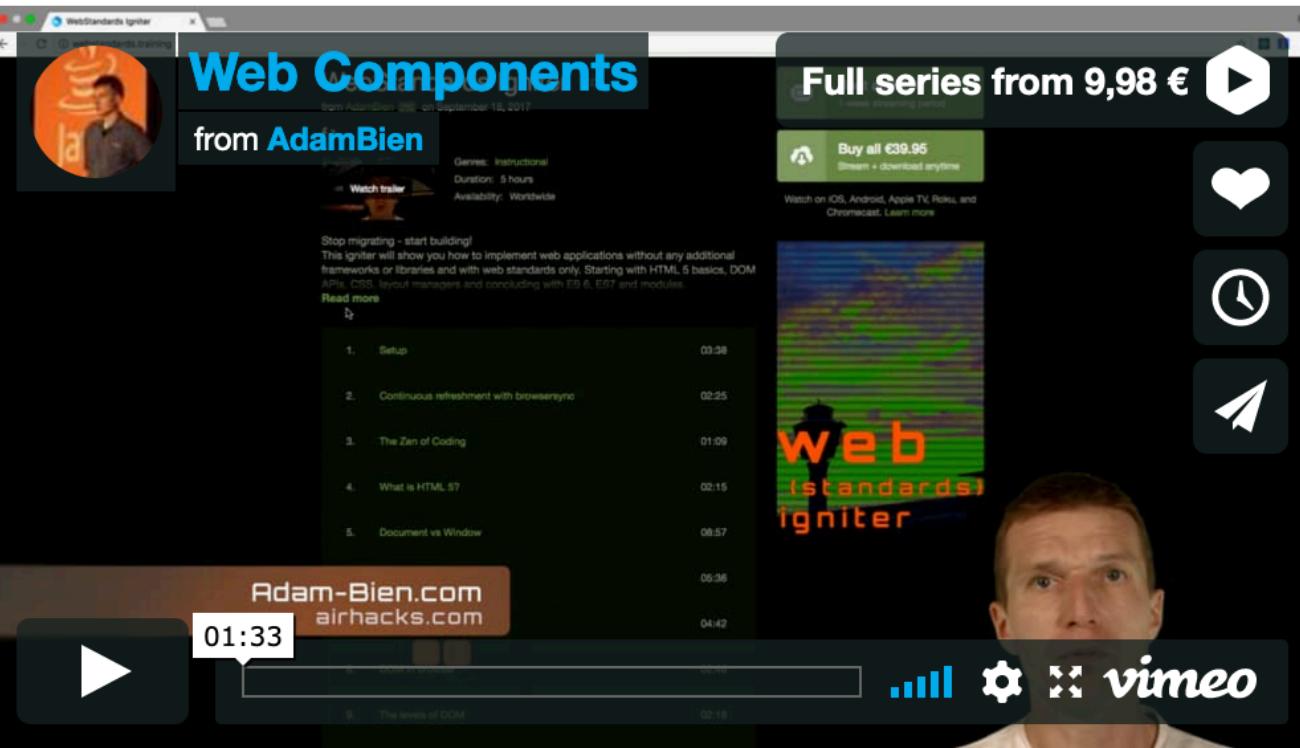
>>web workshops

Web Components, lit-html and redux



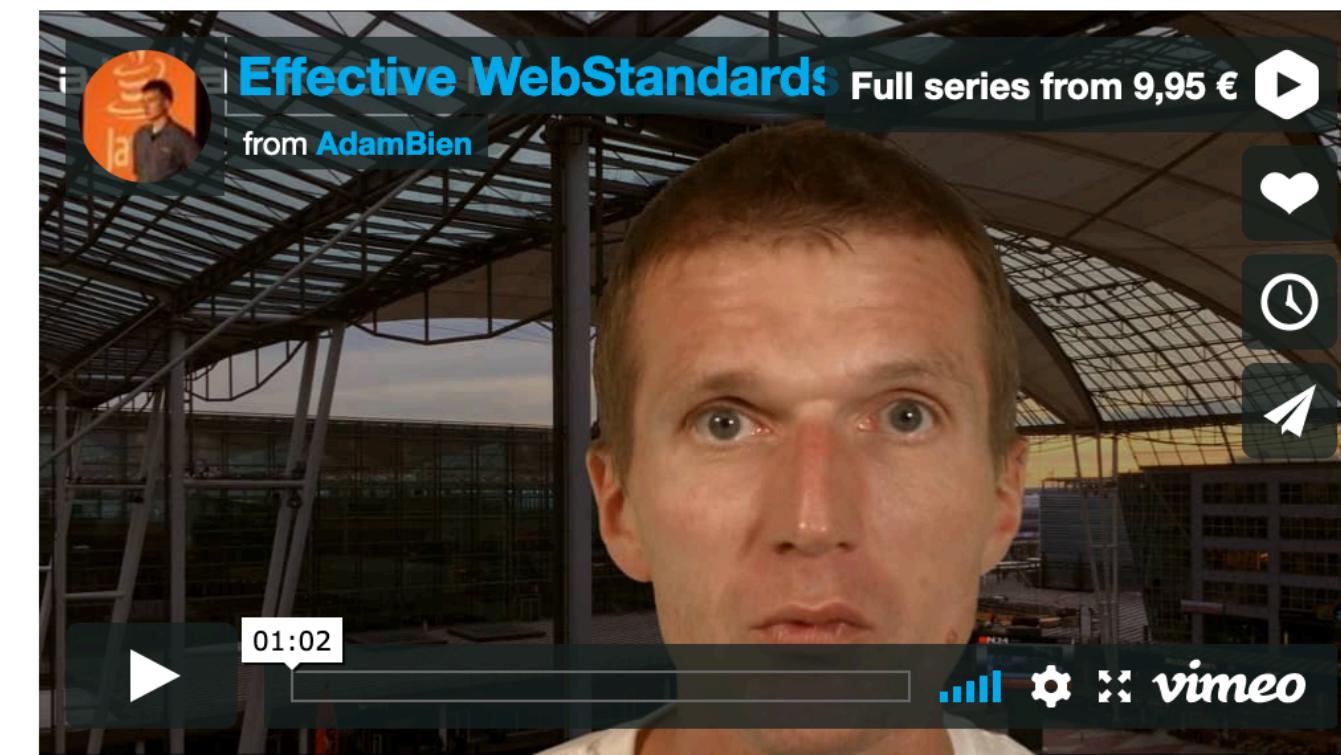
webcomponents-with-redux.training

Web Components Igniter



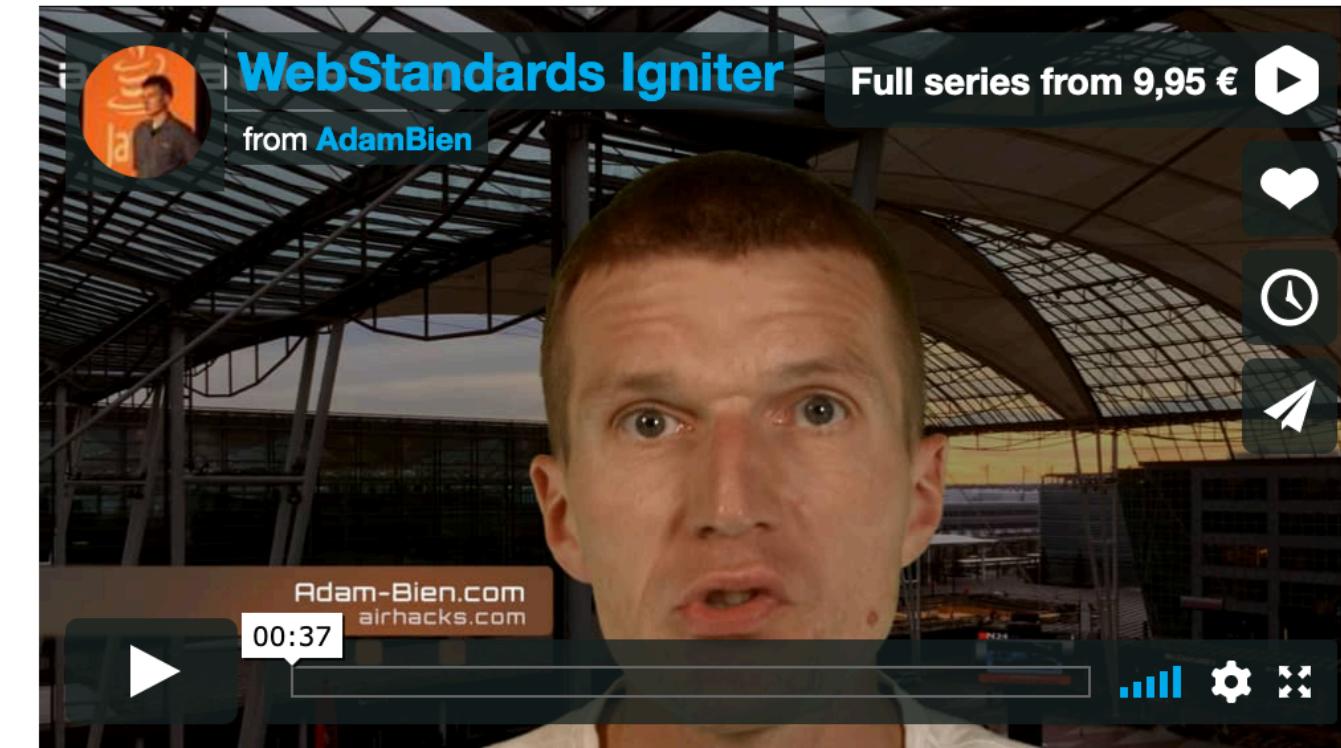
webcomponents.training

Effective Web Apps with Web Standards (only)



effectiveweb.training

WebStandards Igniter



webstandards.training

n.com

>>java workshops

Apps with MicroProfile



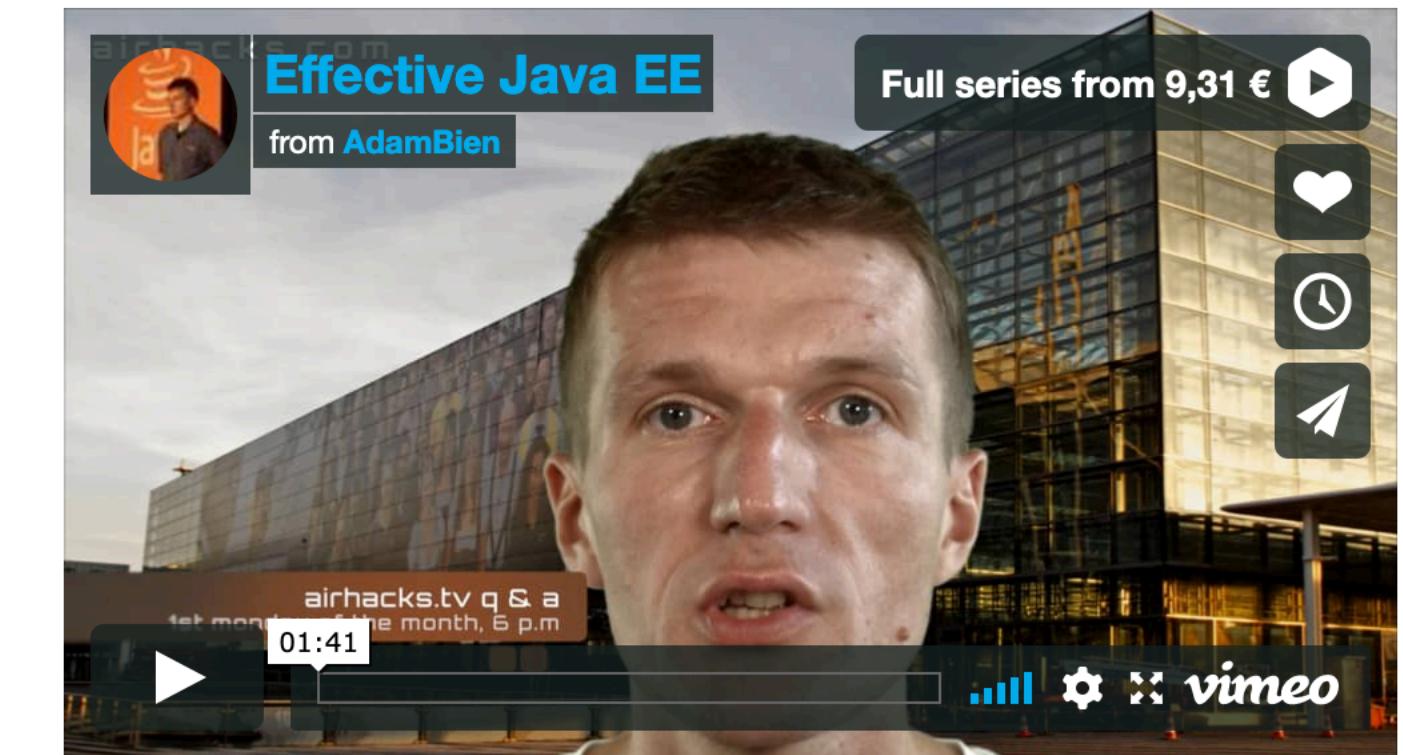
micropatterns.training

Java EE 7 bootstrap



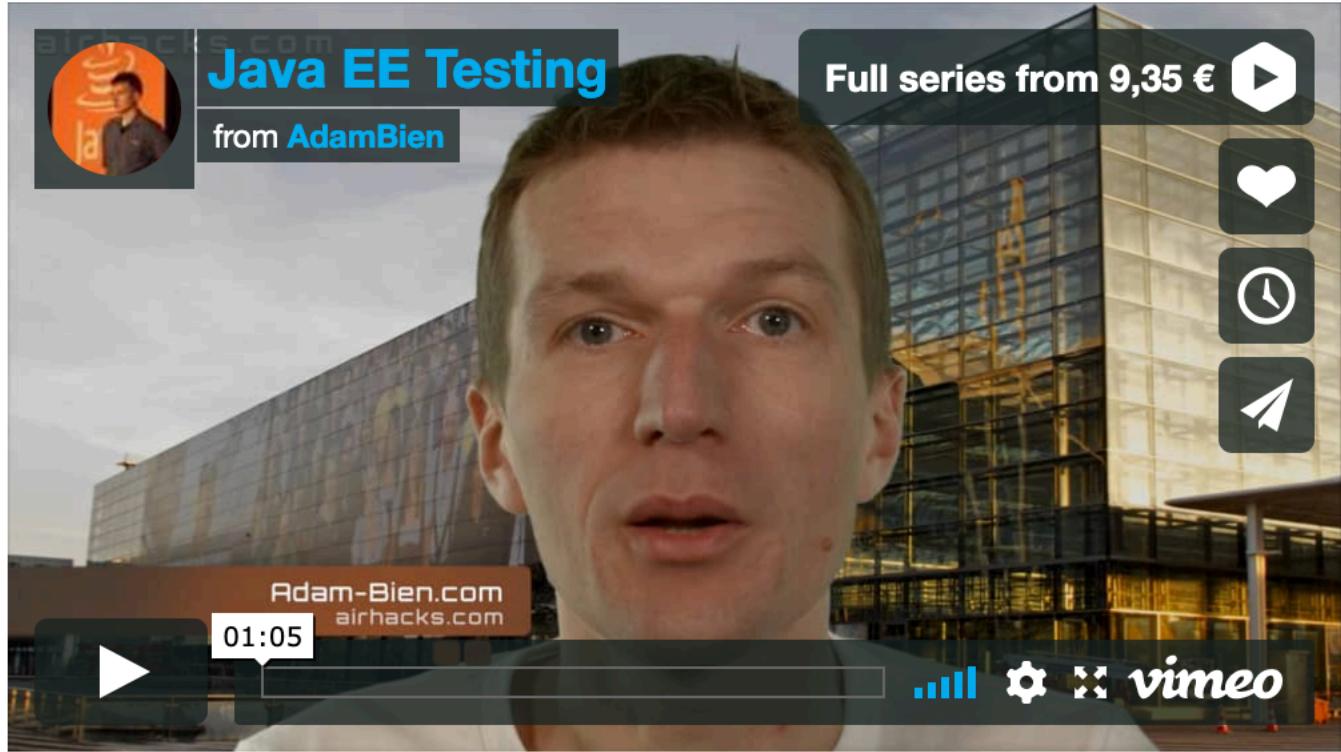
javaeebootstrap.com

Effective Java EE 7



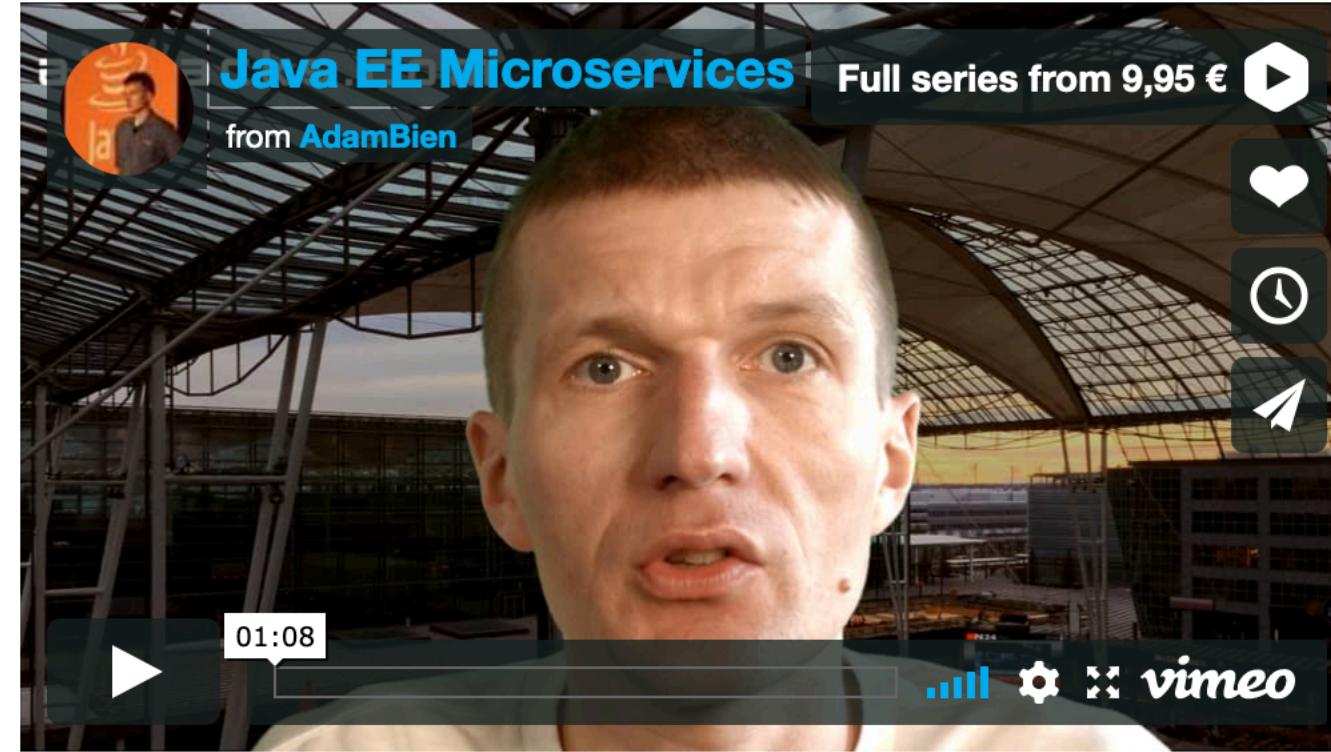
effectivejavaee.com

Java EE 7 Testing



javaeetesting.com

Java EE 7 Microservices



javaeemicro.services

Software Architecture

Software architecture refers to the fundamental structures of a software system and the discipline of creating such structures and systems. Each structure comprises software elements, relations among them, and properties of both elements and relations. (...) It functions as a blueprint for the system and the developing project, laying out the tasks necessary to be executed by the design teams.

Software architecture is about making fundamental structural choices that are costly to change once implemented.

Software architecture choices include specific structural options from possibilities in the design of the software.

Software Design

Software design may refer to either "all the activity involved in conceptualizing, framing, implementing, commissioning, and ultimately modifying complex systems"

or

"the activity following requirements specification and before programming, as ... a stylized software engineering process."

Java?

# Ranking	Programming Language	Percentage (Change)	Trend
1	JavaScript	18.789% (-1.133%)	
2	Python	16.108% (-1.694%)	
3	Java	10.731% (+0.250%)	
4	Go	8.922% (+1.006%)	
5	C++	7.636% (+0.383%)	
6	TypeScript	7.334% (+1.919%)	^
7	Ruby	6.492% (+0.196%)	▼
8	PHP	5.198% (-0.318%)	▼
9	C#	3.797% (-0.204%)	
10	C	3.320% (+0.130%)	
11	Shell	2.011% (-0.337%)	
12	Scala	1.724% (+0.280%)	
13	Rust	1.113% (+0.398%)	^
14	Swift	0.744% (-0.359%)	▼
15	Kotlin	0.670% (+0.014%)	
16	Dart	0.607% (+0.231%)	↗
17	Objective-C	0.409% (-0.118%)	▼
18	Groovy	0.393% (-0.050%)	
19	Elixir	0.377% (-0.071%)	▼
20	DM	0.344% (-0.054%)	▼

Sep 2020	Sep 2019	Change	Programming Language	Ratings	Change
1	2	▲	C	15.95%	+0.74%
2	1	▼	Java	13.48%	-3.18%
3	3		Python	10.47%	+0.59%
4	4		C++	7.11%	+1.48%
5	5		C#	4.58%	+1.18%
6	6		Visual Basic	4.12%	+0.83%
7	7		JavaScript	2.54%	+0.41%
8	9	▲	PHP	2.49%	+0.62%
9	19	▲	R	2.37%	+1.33%
10	8	▼	SQL	1.76%	-0.19%
11	14	▲	Go	1.46%	+0.24%
12	16	▲	Swift	1.38%	+0.28%
13	20	▲	Perl	1.30%	+0.26%
14	12	▼	Assembly language	1.30%	-0.08%
15	15		Ruby	1.24%	+0.03%
16	18	▲	MATLAB	1.10%	+0.04%
17	11	▼	Groovy	0.99%	-0.52%
18	33	▲	Rust	0.92%	+0.55%
19	10	▼	Objective-C	0.85%	-0.99%
20	24	▲	Dart	0.77%	+0.13%

polyglot programming

GraalVM

Jakarta EE?

Specifications

Jakarta EE Platform

The Jakarta EE Platform defines a standard platform for ...

Jakarta EE Web Profile

The Jakarta EE Web Profile defines a profile of the ...

Jakarta Activation

Jakarta Activation defines a set of standard services to: ...

Jakarta Annotations

Jakarta Annotations defines a collection of annotations ...

Jakarta Authentication

Jakarta Authentication defines a general low-level SPI ...

Jakarta Authorization

Jakarta Authorization defines a low-level SPI for ...

Jakarta Batch

Jakarta Batch specifies a Java API plus an XML-based job ...

Jakarta Bean Validation

Jakarta Bean Validation provides an object level ...

Jakarta Concurrency

Jakarta Concurrency provides a specification for using ...

Jakarta Connectors

The Jakarta Connectors specification defines a standard ...

Jakarta Contexts and Dependency Injection

Jakarta Contexts and Dependency Injection defines a ...

Jakarta Debugging Support for Other Languages

Jakarta Debugging Support for Other Languages provides a ...

Jakarta Dependency Injection

Jakarta Dependency Injection specifies a means for ...

Jakarta Deployment

Jakarta Deployment defines standard APIs that will enable ...

Jakarta Enterprise Beans

Jakarta Enterprise Beans defines an architecture for the ...

Jakarta Enterprise Web Services

This specification defines the Web Services for Jakarta ...

Jakarta Expression Language

Jakarta Expression Language defines an expression ...

Jakarta Interceptors

Jakarta Interceptors defines a means of interposing on ...

Jakarta JSON Binding

Jakarta JSON Binding defines a binding framework for ...

Jakarta JSON Processing

Jakarta JSON Processing defines a Java(R) based framework ...

Jakarta Mail

Jakarta Mail defines a platform-independent and ...

Jakarta Managed Beans

Jakarta Managed Beans defines a set of basic services for ...

Jakarta Management

Jakarta Management defines a standard management model ...

Jakarta Messaging

Jakarta Messaging describes a means for Java applications ...

Jakarta MVC

Jakarta MVC defines a standard for creating web ...

Jakarta NoSQL

Jakarta NoSQL defines a standard for creating ...

Jakarta Persistence

Jakarta Persistence defines a standard for management of ...

Jakarta RESTful Web Services

Jakarta RESTful Web Services provides a foundational API ...

Jakarta Security

Jakarta Security defines a standard for creating secure ...

Jakarta Server Faces

Jakarta Server Faces defines an MVC framework for ...

Jakarta Server Pages

Jakarta Server Pages defines a template engine for web ...

Jakarta Servlet

Jakarta Servlet defines a server-side API for handling ...

Jakarta SOAP with Attachments

Jakarta SOAP with Attachments defines an API enabling ...

Jakarta Standard Tag Library

Jakarta Standard Tag Library represents a set of tags to ...

Jakarta Transactions

Jakarta Transactions defines a standard that allows the ...

Jakarta Web Services Metadata

Jakarta Web Services Metadata defines a programming model ...

Jakarta WebSocket

Jakarta WebSocket defines a API for Server and Client ...

Jakarta XML Binding

The Jakarta XML Binding provides an API and tools that ...

Jakarta XML Registries

Jakarta XML Registries describes Java API's designed ...

Jakarta XML RPC

Jakarta XML-based RPC defines consistent Java APIs for ...

Jakarta XML Web Services Specification

Jakarta XML Web Services defines a means for implementing ...

Jakarta EE 8 Full Platform Compatible Products

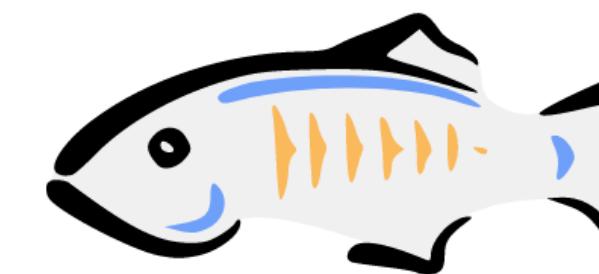


Apusic AAS
Kingdee Apusic cloud computing
Co., Ltd.

Version: V10

[Download](#)

[Proof of compatibility](#)



Eclipse Glassfish
Eclipse Foundation

Version: 5.1.0, Full Profile

[Download](#)

[Proof of compatibility](#)



**JBoss Enterprise
Application Platform**
Red Hat

Version: 7.3.0.GA

[Download](#)

[Proof of compatibility](#)



JEUS
TmaxSoft Co., Ltd

Version: 8.5

[Download](#)

[Proof of compatibility](#)



Open Liberty
IBM Corporation

Version: 19.0.0.6

[Download](#)

[Proof of compatibility](#)



Payara Server
Payara Services Limited

Version: 5.193.1

[Download](#)

[Proof of compatibility](#)



Primeton AppServer
Primeton

Version: 7

[Download](#)

[Proof of compatibility](#)



WildFly
Red Hat

Version: 18.0.0.Final

[Download](#)

[Proof of compatibility](#)

no politics

MicroProfile?

Open
Tracing 1.3

Open API 1.1

Rest Client 1.4

Config 1.4

Fault
Tolerance 2.1

Metrics 2.3

JWT
Propagation 1.1

Health 2.2

CDI 2.0

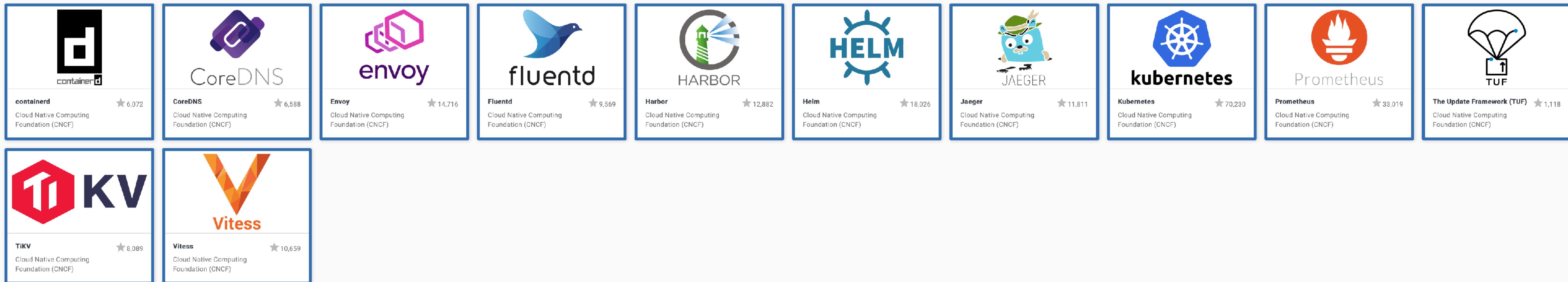
JSON-P 1.1

JAX-RS 2.1

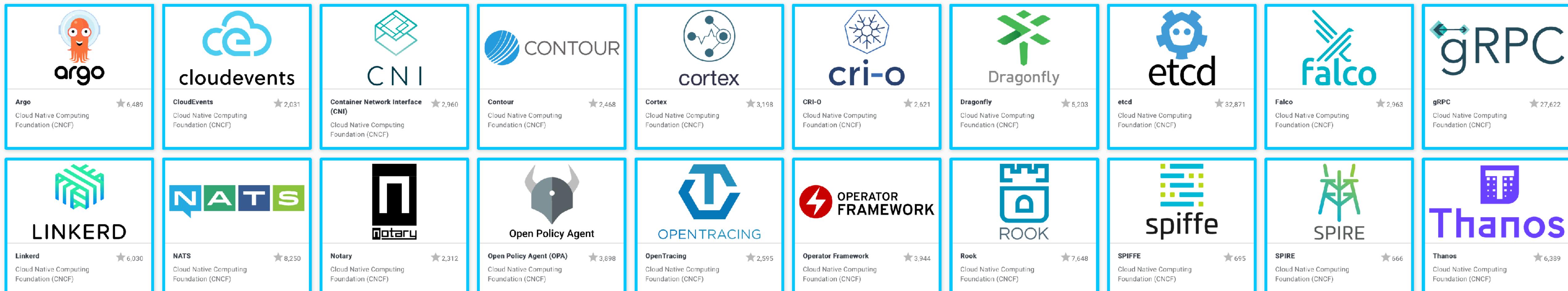
JSON-B 1.0

cncf.io

CNCF Graduated Projects (12)



CNCF Incubating Projects (20)



API <-> SPI and RI

specs vs. frameworks

commercial support?

crisis: back to standards

decision fatigue

"In decision making and psychology, decision fatigue refers to the deteriorating quality of decisions made by an individual after a long session of decision making. It is now understood as one of the causes of irrational trade-offs in decision making. Decision fatigue may also lead to consumers making poor choices with their purchases."

https://en.wikipedia.org/wiki/Decision_fatigue

productivity

Agile?

Cargo Cult?

"Cargo cult programming is a style of computer programming characterized by the ritual inclusion of code or program structures that serve no real purpose."

[http://en.wikipedia.org/wiki/Cargo_cult_programming]

Information Cascades

“Informational (or information) cascades happen when Internet users start passing on information they assume to be true, but cannot know to be true, based on information on what other users are doing. (...). This type of information spreading is hard to stop. Information cascades over social media and the Internet may also be harmless, and may contain truthful information.”

[https://en.wikipedia.org/wiki/Information_cascade]

Parkinson's Law of Triviality

Documentation

JavaDoc + OpenAPI

Testing

Meta: How To Start a Project?

Understanding CAP

**CAP vs. FLP (Fischer,
Lynch and Paterson)**

In an asynchronous network where messages may be delayed but not lost, there is no consensus algorithm that is guaranteed to terminate in every execution for all starting conditions, if at least one node may fail-stop.

Atomic Broadcast

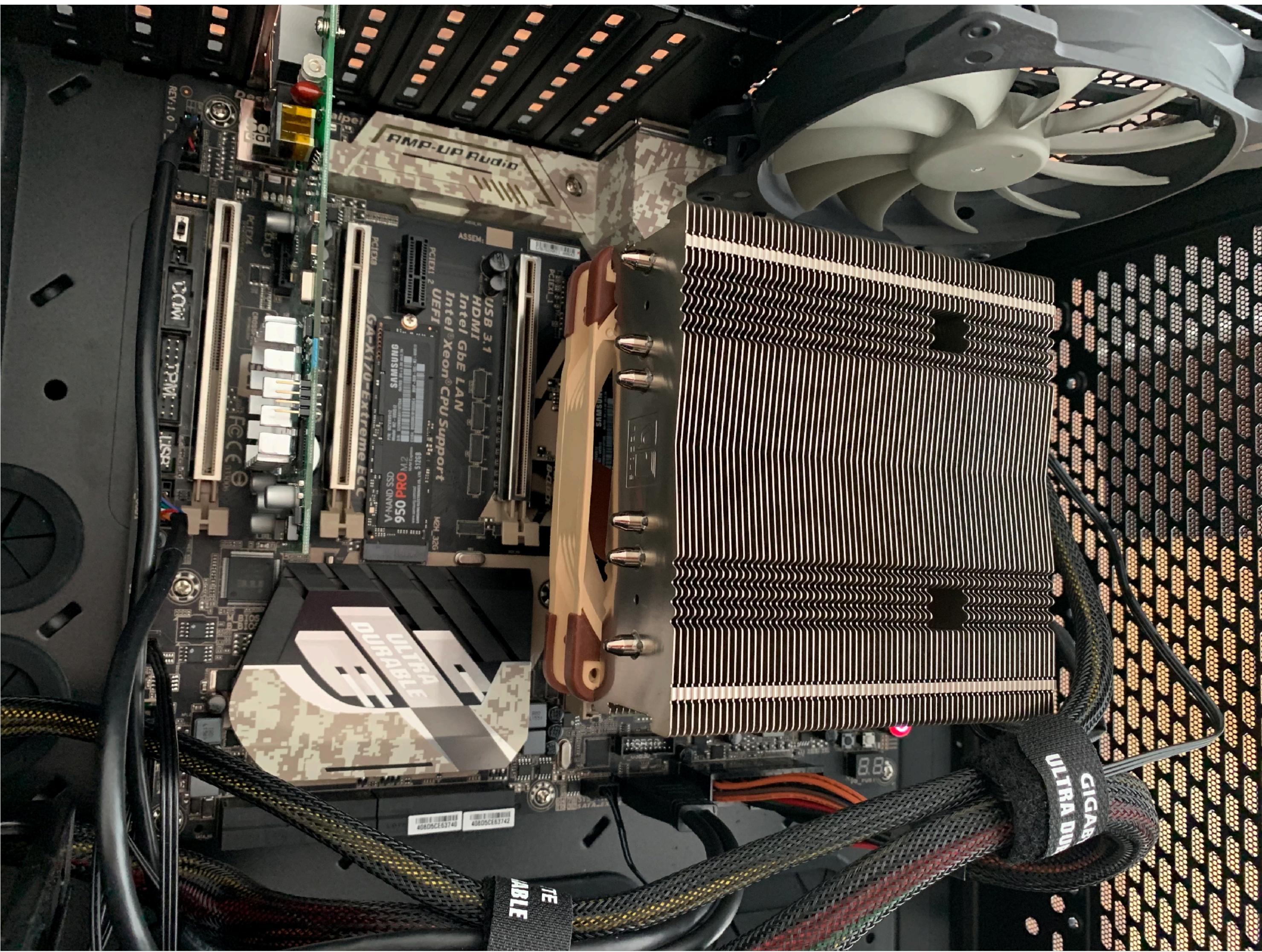
In fault-tolerant distributed computing, an atomic broadcast or total order broadcast is a broadcast where all correct processes in a system of multiple processes receive the same set of messages in the same order; that is, the same sequence of messages. The broadcast is termed "atomic" because it either eventually completes correctly at all participants, or all participants abort without side effects. Atomic broadcasts are an important distributed computing primitive

Hardware Costs

Sun E10K 	E10K Server - Enterprise 10000 Server - Sun Microsystems Sun Enterprise E10000 Server. This is a complete server with all system boards, power supplies, fan trays etc. It was installed at AOL and was in good working order until it was deinstalled. The CPU's and memory have been removed, but this is a great machine for parts. If you are supporting E10K servers, this is a perfect spare. Click here for technical information on the Sun Fire E10K server <ul style="list-style-type: none">• 30-Day Warranty on all hardware• Great for spare parts. Complete list of installed options is available upon request.• Can ship today• in stock. Call for availability on quantities over 1				
Delta Marketing Group, Inc. <small>since 1986</small> We would also be happy to take your order or answer any questions over the phone 480-367-6698 For Sales dial 1 or Michael Hankerson extension 101	Manufactured By: Sun Microsystems  Print Product Description  Email a Friend  Sign-up for Monthly Specials  Update My Profile <table border="1"><tr><td>List Price:\$1,200,000.00</td></tr><tr><td>Qty <input type="text" value="1"/> qty:</td></tr><tr><td>Add To Cart</td></tr><tr><td>\$1,995.00</td></tr></table>	List Price:\$1,200,000.00	Qty <input type="text" value="1"/> qty:	Add To Cart	\$1,995.00
List Price:\$1,200,000.00					
Qty <input type="text" value="1"/> qty:					
Add To Cart					
\$1,995.00					

home grown





Who needs 24*7?



Store

Mac

iPod

iPhone

iPad

iTunes

Support



We'll be back.

We're busy updating the Apple Store
for you and will be back soon.



Store

Mac

iPod

iPhone

iPad

iTunes

Support



Wir sind bald wieder da.

Wir aktualisieren gerade das Angebot
im Apple Online Store. Die Seite ist
demnächst wieder verfügbar.



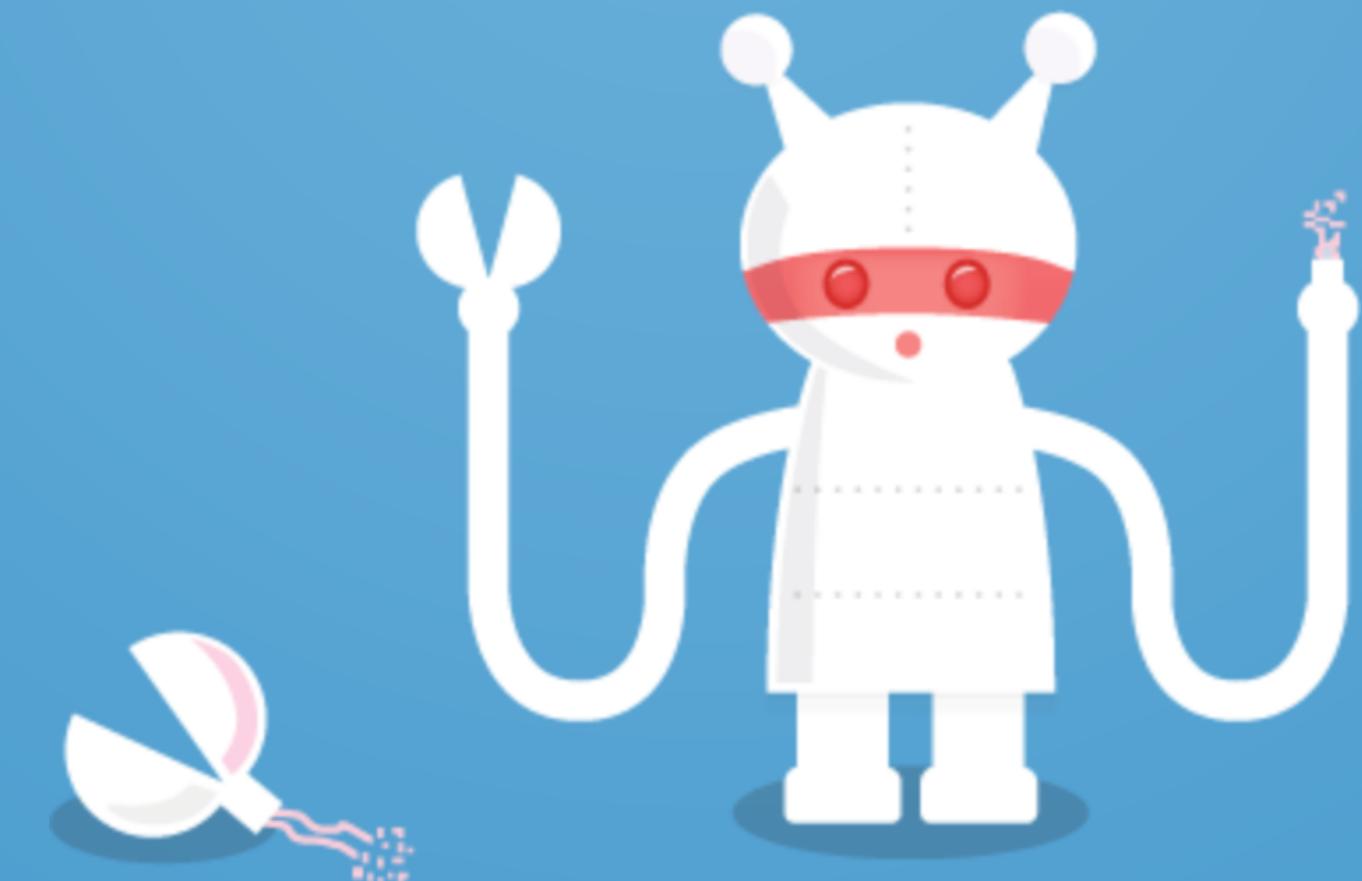
Home →

Something is technically wrong.

Thanks for noticing—we're going to fix it up and have things back to normal soon.

العربية Dansk Deutsch English English UK Español فارسی Suomi Filipino Français עברית हिन्दी Magyar
Bahasa Indonesia Italiano 日本語 한국어 Bahasa Melayu Nederlands Norsk Polski Português Русский Svenska
ภาษาไทย Türkçe 简体中文

© 2016 Twitter About Help Status



...and

[Store](#)[Mac](#)[iPod](#)[iPhone](#)[iPad](#)[iTunes](#)[Support](#)

Apple Store

[1-800-MY-APPLE](#)[Account](#)[Cart](#)

We're sorry.

We're sorry, but our Order Status services are extremely busy right now. Please try again later.

Apple uses industry-standard encryption to protect the confidentiality of your personal information.

**BASE vs.ACID, or how
to start a project**

Clear Code

Design

DRY and DIE

KISS

YAGNI

single responsibility

A class should only have a single responsibility, that is, only changes to one part of the software's specification should be able to affect the specification of the class.

open-closed

In object-oriented programming, the open–closed principle states "software entities (classes, modules, functions, etc.) should be open for extension, but closed for modification"; that is, such an entity can allow its behaviour to be extended without modifying its source code.

liskov substitution

Objects in a program should be replaceable with instances of their subtypes without altering the correctness of that program.

interface segregation

In the field of software engineering, the interface-segregation principle (ISP) states that no client should be forced to depend on methods it does not use. ISP splits interfaces that are very large into smaller and more specific ones so that clients will only have to know about the methods that are of interest to them. Such shrunken interfaces are also called role interfaces.

dependency inversion

One should "depend upon abstractions, [not] concretions." [4]

=> SOLID

package principles

In computer programming, package principles are a way of organizing classes in larger systems to make them more organized and manageable. They aid in understanding which classes should go into which packages (package cohesion) and how these packages should relate with one another (package coupling).

https://en.wikipedia.org/wiki/Package_principles

adam-bien.com

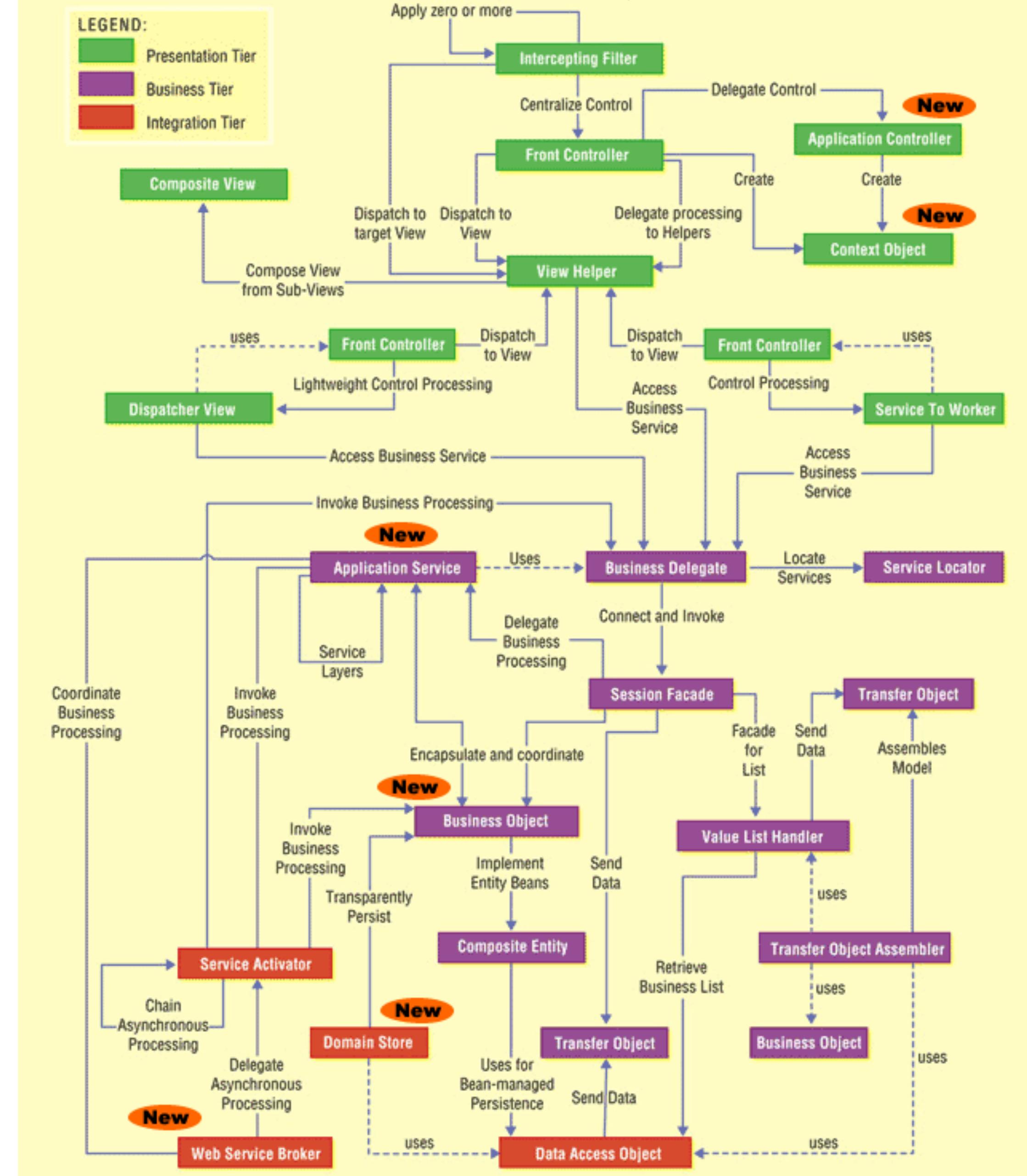
what about reuse?

the most important thing is...

...discipline

Deprecated J2EE Patterns

Core J2EE Patterns, 2nd Edition



GoF Patterns

Factory, Abstract Factory, Singleton, Builder,
Prototype, Adapter, Bridge, Decorator, Facade,
Flyweight, Proxy, Template, Observer,
Command, Strategy, State

VMs vs. containers

Immutable Infrastructure

kubernetes

observability

In control theory, observability is a measure of how well internal states of a system can be inferred from knowledge of its external outputs.

clouds

cloud native

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds.

Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

private vs. public

Serverless / FaaS

service meshes

- Envoy, Linkerd, Traefik
- Proxies
- Out-of-the-box Robustness Patterns
- Universal Data Plane API

Conway's Law

GRASP

- High Cohesion, Minimal Coupling
- Protected Variations
- Information Expert
- Polymorphism
- Pure Fabrication

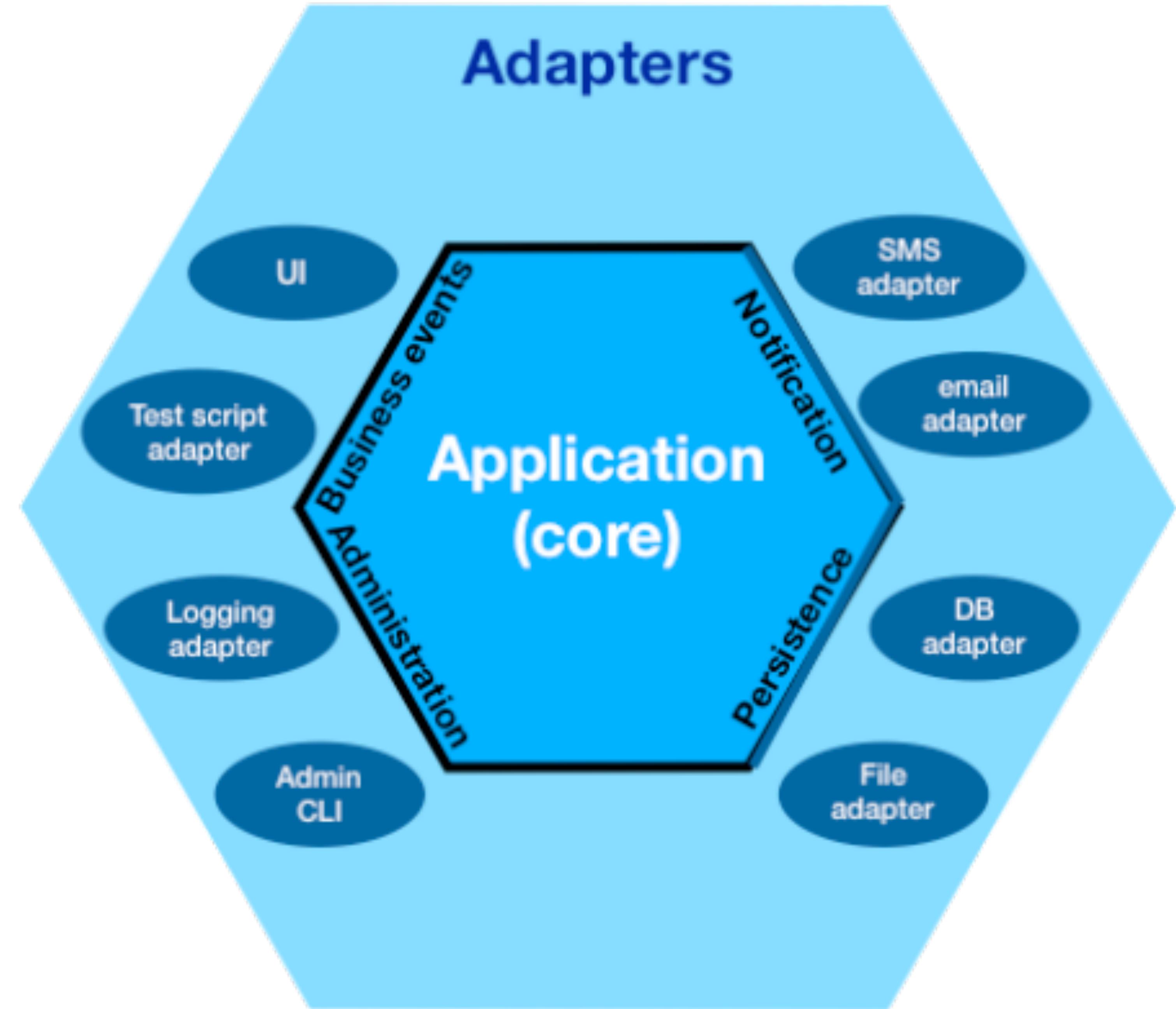
request - response

coupling?

monoliths

Hexagonal Architecture

[<http://wiki.c2.com/?HexagonalArchitecture>]



BCE

Designing plugins,
templates and hooks

RESTful architectures vs. SOAP

Stateful vs. stateless Java architectures

- Where is the state?
- What about the scalability?

Caveats of asynchronous applications

microservices

robustness patterns

- Timeout: Define a duration for timeout
- Retry: Define a criteria on when to retry
- Fallback: provide an alternative solution for a failed execution.
- CircuitBreaker: offer a way of fail fast by automatically failing execution to prevent the system overloading and indefinite wait or timeout by the clients.
- Bulkhead: isolate failures in part of the system while the rest part of the system can still function.

Service governance and versioning

- API Versioning
- Dealing with extensions
- (In)Compatibilities
- SCM

Deployment Strategies

- Blue-Green Deployment
- Hot Deployment
- Virtualization
- Containers
- Canary Deployment

Pragmatic modularization of Java applications

Structuring

- The Vision
- Cohesion <-> Coupling
- Essential Entities and Business Components
- “Maximal Cohesion, Minimal Coupling”

Domain Event

**“Captures the memory of something interesting
which affects the domain”**

[<https://martinfowler.com/eaaDev/DomainEvent.html>]

Event Naming

- subject - verb - object
- “reader clicks on post” -> “reader clicked on post” (at timestamp)

Event Structure

```
{  
  "name": "READER_CLICKED_POST",  
  "reader": {  
    "id": 42,  
    "name": "duke"  
  },  
  "post": {  
    "title": "Jakarta EE Rocks",  
    "uri": "http://adam-bien.blog/..."  
  },  
  "timestamp": "2020-01-10"  
}
```

Command Query Responsibility Segregation (CQRS)

“CQRS states that every method should either be a command that performs an action, or a query that returns data to the caller, but not both.

In other words, asking a question should not change the answer.“

[http://en.wikipedia.org/wiki/Command-query_separation]

Event Sourcing

“Event Sourcing ensures that all changes to application state are stored as a sequence of events. Not just can we query these events, we can also use the event log to reconstruct past states, and as a foundation to automatically adjust the state to cope with retroactive changes.”

[<http://martinfowler.com/eaaDev/EventSourcing.html>]

MoM

- API Compatibility
- Deliver Once And Only Once
- Persistent Queues / Durable Subscribers
- Poisoned Messages

Reactive Programming

tranceivers

“In radio communication, a transceiver is a device that is able to both transmit and receive information through a transmission medium. It is a combination of a transmitter and a receiver, hence the name transceiver. Transmission is usually accomplished via radio waves, but communications satellites, wired connections, and optical fiber systems can also be used.”

[<https://en.wikipedia.org/wiki/Transceiver>]

Logging, errorhandling

- Who needs logs?
- Error classification
- Audits and traces
- Exceptions and status codes

Management and monitoring

- JMX, REST
- "Business monitoring"
- Automated setup and deployment

persistence

Decide early what is more important to you: the structure of the database or your code's design. Otherwise, you will waste a lot of time, and one of them will suffer anyway.

RDBMS

Non-Relational DBs (NoSQL)

NoSQL

- Key Value
- Document
- Graph

Continuous Integration and Continuous Deployment

data lakes

“A data lake is a system or repository of data stored in its natural/raw format,[1] usually object blobs or files. A data lake is usually a single store of all enterprise data including raw copies of source system data and transformed data used for tasks such as reporting, visualization, advanced analytics and machine learning.”

[https://en.wikipedia.org/wiki/Data_lake]

workshops.adam-bien.com

NEW Virtual Online Workshops, Winter 2020:

MicroProfile with Quarkus, December 10th, 2020

**Micro Frontends with Web Components, lit-html and redux,
December 11th, 2020**

[past workshops / catalog](#)

save the date: March, 23rd, 2021

building apps

with

streams

events

&

logs

messages

airhacks.live

Thank You!

twitter.com/AdamBien