

# Software Quality For Startups

Cyrille Savelief, CSO, MNCC

Functionality

Usability

Reliability

Performance

Supportability

# Who

## Cyrille SAVELIEF



Graduated from the Parisian Master of Research in Computer Science in 2007. Worked 4 years at FactSet Research Systems as software engineer for the real-time data integration and distribution team. Since 2011, CSO at MNCC, a performance-focused Web startup.

# Overview



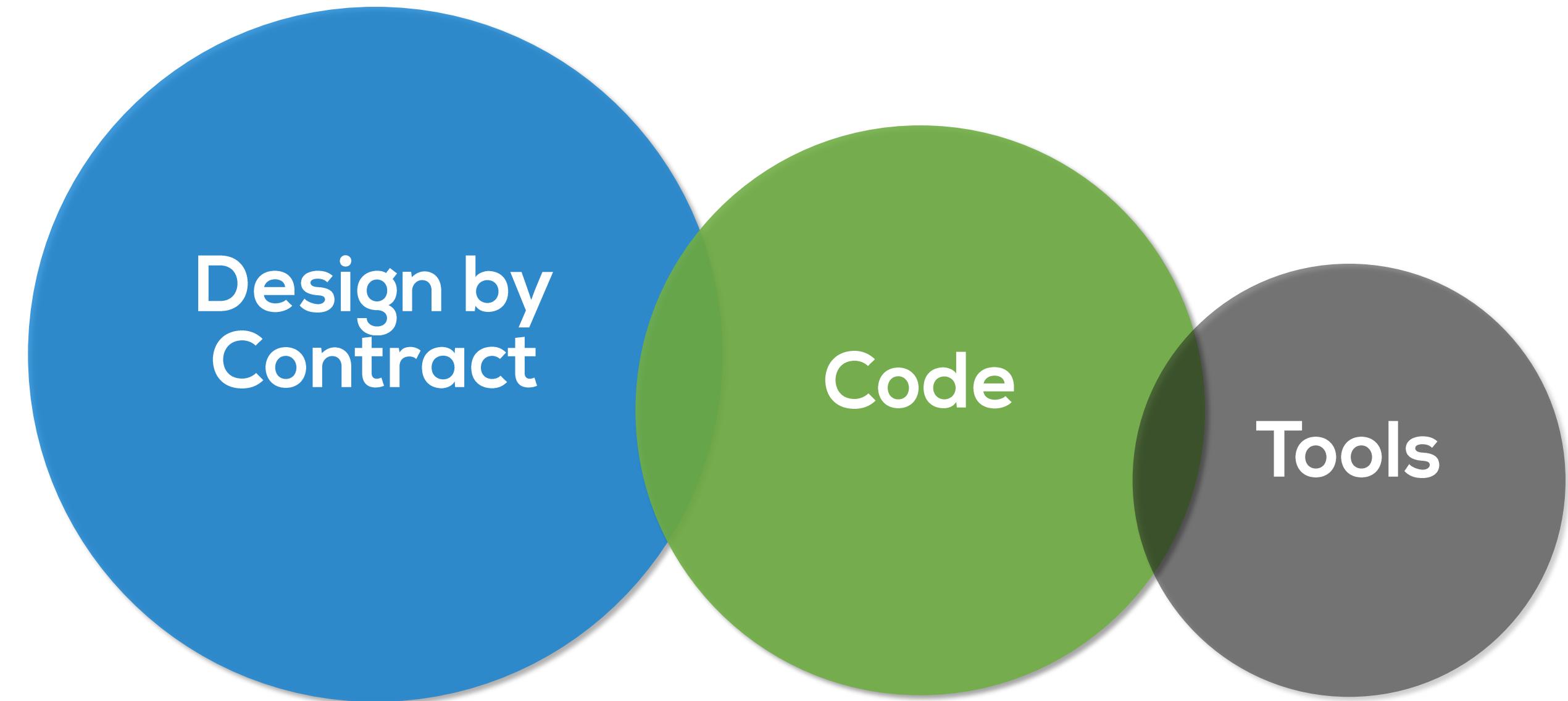
## Quality Attributes of Web Applications

Designing and building Web applications that scale well represents one of today's most interesting and most important software development challenges.

- Reliability
- Usability
- Security
- Availability
- Scalability
- Maintainability
- Time-to-market

# A

Agenda





“ ”

# J.WEBBER

The fastest code is that  
which is neither downloaded  
nor ran.

# Web Dev.

Angular.js,  
Backbone.js,  
Etc.

Mustache.js,  
Handlebar.js,  
Etc.

Less, sass, etc.

Uglify.js, closure  
compiler, etc.



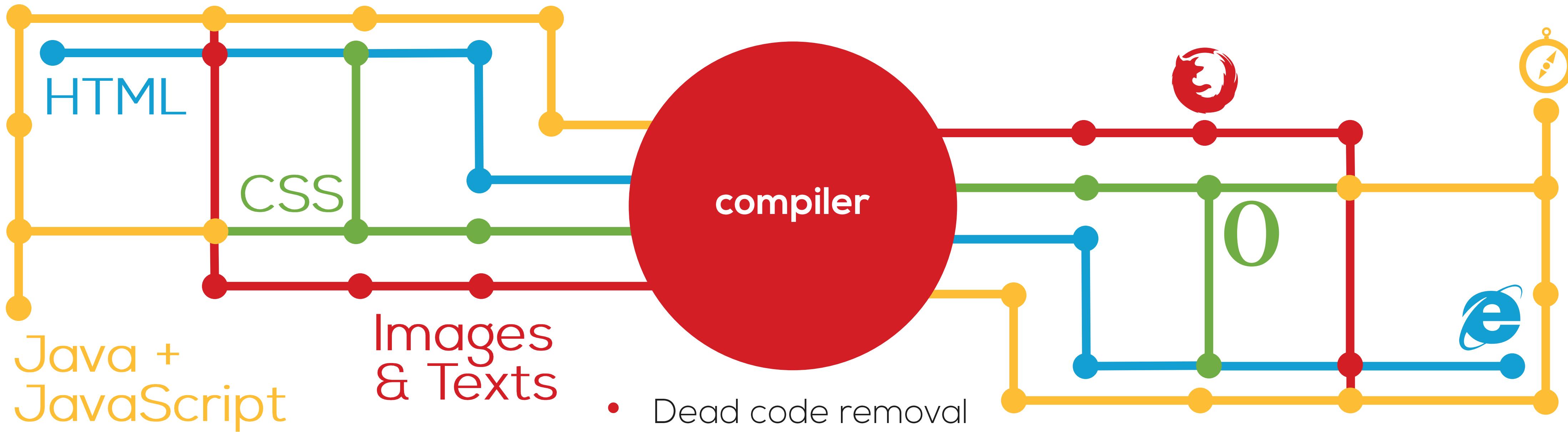


“ ”

# R.CROMWELL

Better one tool than many.

# GWT



- Dead code removal
- Inlining
- Renaming
- Zipping
- Perfect caching
- Deferred binding

# Example

## SQUARE.JS

---

```
FUNCTION SQUARE ( LENGTH ) {  
    THIS.LENGTH = LENGTH;  
}  
  
SQUARE.PROTOTYPE.GETAREA () {  
    RETURN THIS.LENGTH * THIS.LENGTH;  
}  
  
SQUARE.PROTOTYPE.GETPERIMETER () {  
    RETURN 4 * THIS.LENGTH;  
}
```

## CIRCLE.JS

---

```
FUNCTION CIRCLE ( RADIUS ) {  
    THIS.RADIUS = RADIUS;  
}  
  
CIRCLE.PROTOTYPE.GETAREA () {  
    RETURN MATH.PI * THIS.RADIUS * THIS.RADIUS;  
}  
  
CIRCLE.PROTOTYPE.GETPERIMETER () {  
    RETURN 2 * MATH.PI * THIS.RADIUS;  
}
```

# Example

## 1. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );  
VAR AREA = C.GETAREA ();
```

# Example

## 1. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );  
VAR AREA = C.GETAREA ();
```

## 2. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );  
VAR AREA = C.RADIUS * C.RADIUS * MATH.PI;
```

# Example

## 1. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );
VAR AREA = C.GETAREA ();
```

## 2. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );
VAR AREA = C.RADIUS * C.RADIUS * MATH.PI;
```

## 3. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );
VAR AREA = 2 * 2 * MATH.PI;
```

# Example

## 1. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );
VAR AREA = C.GETAREA ();
```

## 2. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );
VAR AREA = C.RADIUS * C.RADIUS * MATH.PI;
```

## 3. EXAMPLE.JS

---

```
CIRCLE C = NEW CIRCLE ( 2 );
VAR AREA = 2 * 2 * MATH.PI;
```

## 4. EXAMPLE.JS

---

```
VAR AREA = 12.566370614...;
```

# Example

## SQUARE.JS

---

```
FUNCTION SQUARE ( LENGTH ) {  
    THIS.LENGTH = LENGTH;  
}  
  
SQUARE.PROTOTYPE.GETAREA () {  
    RETURN THIS.LENGTH * THIS.LENGTH;  
}  
  
SQUARE.PROTOTYPE.GETPERIMETER () {  
    RETURN 4 * THIS.LENGTH;  
}
```

## CIRCLE.JS

---

```
FUNCTION CIRCLE ( RADIUS ) {  
    THIS.RADIUS = RADIUS;  
}  
  
CIRCLE.PROTOTYPE.GETAREA () {  
    RETURN MATH.PI * THIS.RADIUS * THIS.RADIUS;  
}  
  
CIRCLE.PROTOTYPE.GETPERIMETER () {  
    RETURN 2 * MATH.PI * THIS.RADIUS;  
}
```

# Example

## SQUARE.JS

---

```
FUNCTION SQUARE ( LENGTH ) {  
    ————— THIS.LENGTH = LENGTH;  
}  
  
SQUARE.PROTOTYPE.GETAREA () {  
    ————— RETURN THIS.LENGTH * THIS.LENGTH;  
}  
  
SQUARE.PROTOTYPE.GETPERIMETER () {  
    ————— RETURN 4 * THIS.LENGTH;  
}
```

## CIRCLE.JS

---

```
FUNCTION CIRCLE ( RADIUS ) {  
    ————— THIS.RADIUS = RADIUS;  
}  
  
CIRCLE.PROTOTYPE.GETAREA () {  
    ————— RETURN MATH.PI * THIS.RADIUS * THIS.RADIUS;  
}  
  
CIRCLE.PROTOTYPE.GETPERIMETER () {  
    ————— RETURN 2 * MATH.PI * THIS.RADIUS;  
}
```

# Example

## SQUARE.JS

---

```
FUNCTION SQUARE ( LENGTH ) {  
    ————— THIS.LENGTH = LENGTH;  
}  
  
SQUARE.PROTOTYPE.GETAREA () {  
    ————— RETURN THIS.LENGTH * THIS.LENGTH;  
}  
  
SQUARE.PROTOTYPE.GETPERIMETER () {  
    ————— RETURN 4 * THIS.LENGTH;  
}
```

## CIRCLE.JS

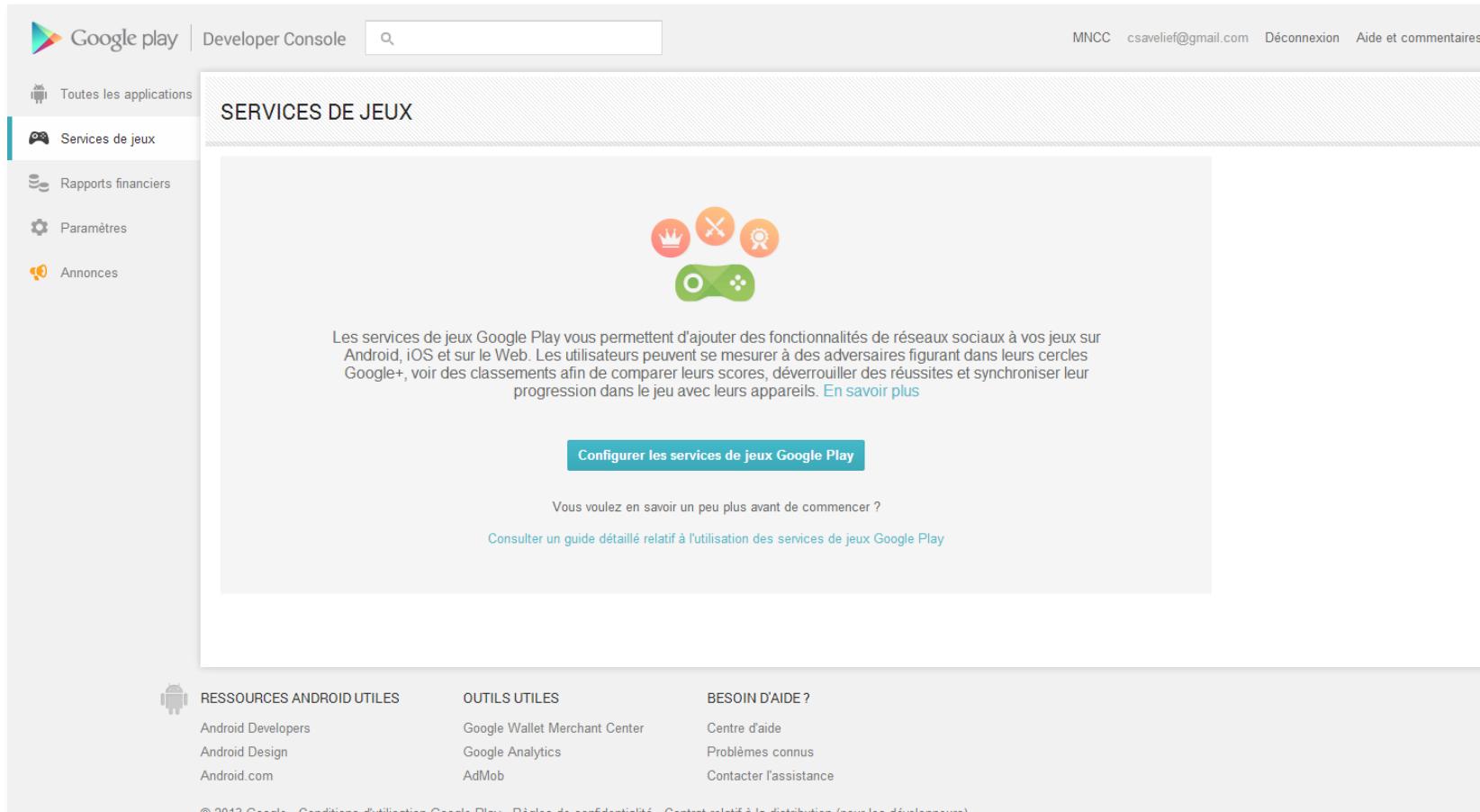
---

```
FUNCTION CIRCLE ( RADIUS ) {  
    ————— THIS.RADIUS = RADIUS;  
}  
  
CIRCLE.PROTOTYPE.GETAREA () {  
    ————— RETURN MATH.PI * THIS.RADIUS * THIS.RADIUS;  
}  
  
CIRCLE.PROTOTYPE.GETPERIMETER () {  
    ————— RETURN 2 * MATH.PI * THIS.RADIUS;  
}
```

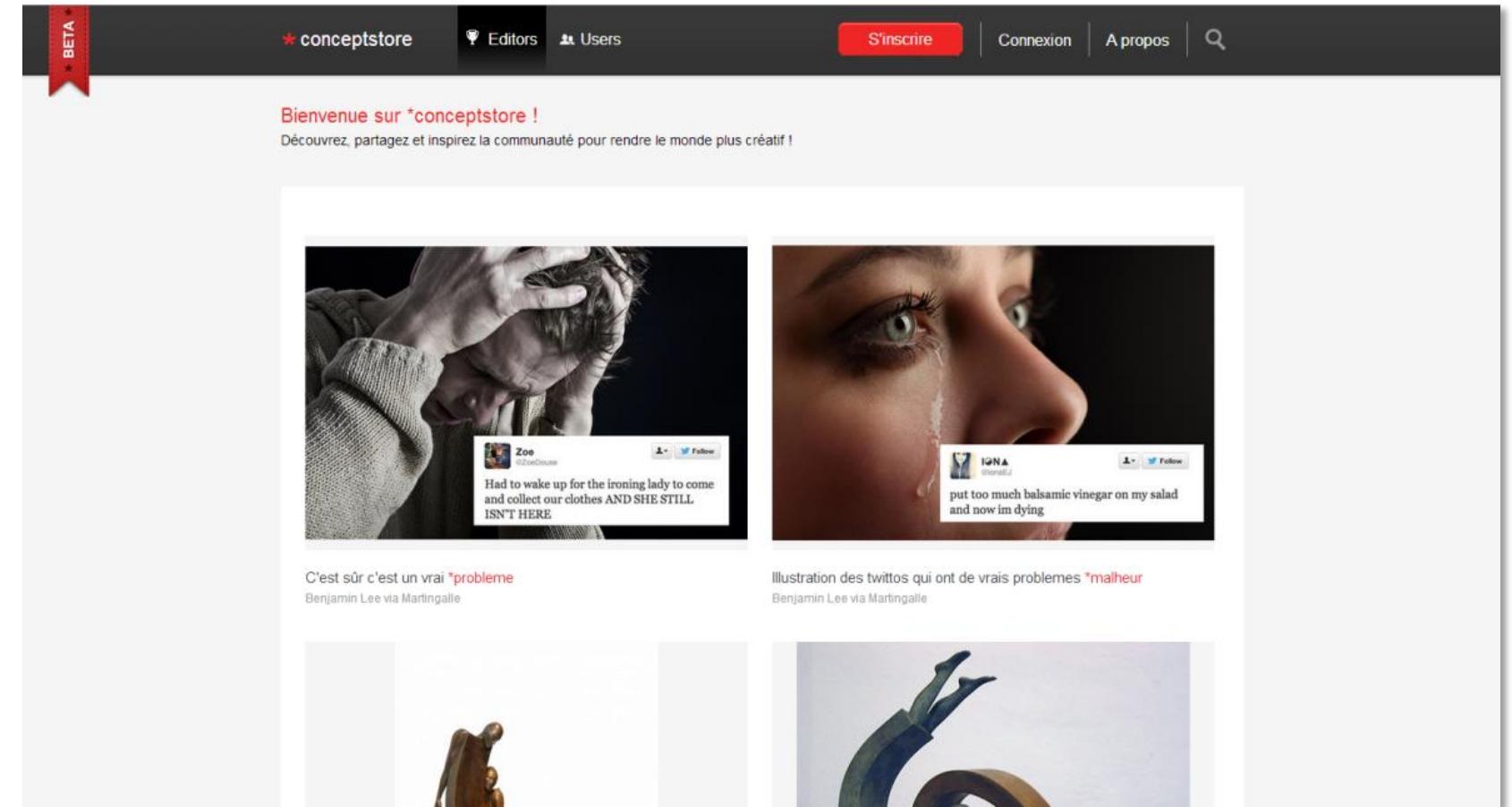
# Example

**VAR AREA = 12.566370614...;**

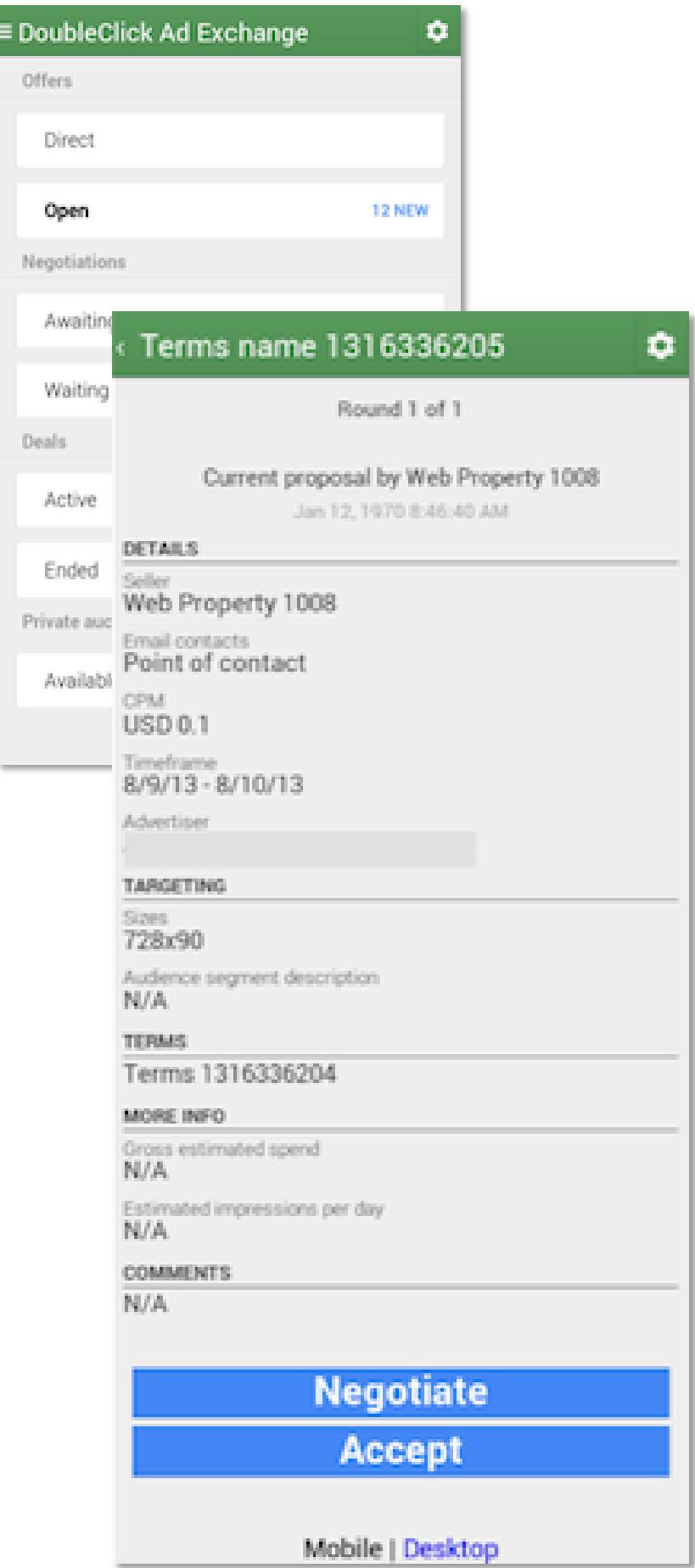
# Who uses GWT?



The screenshot shows the 'SERVICES DE JEUX' section of the Google Play Developer Console. It features a central area with four circular icons representing different service types. Below this is a text block explaining how Google Play services can be added to games. At the bottom, there are links for 'Configurer les services de jeux Google Play' and 'Consulter un guide détaillé'. A footer navigation bar includes links for 'RESSOURCES ANDROID UTILES', 'OUTILS UTILES', and 'BESOIN D'AIDE ?'.



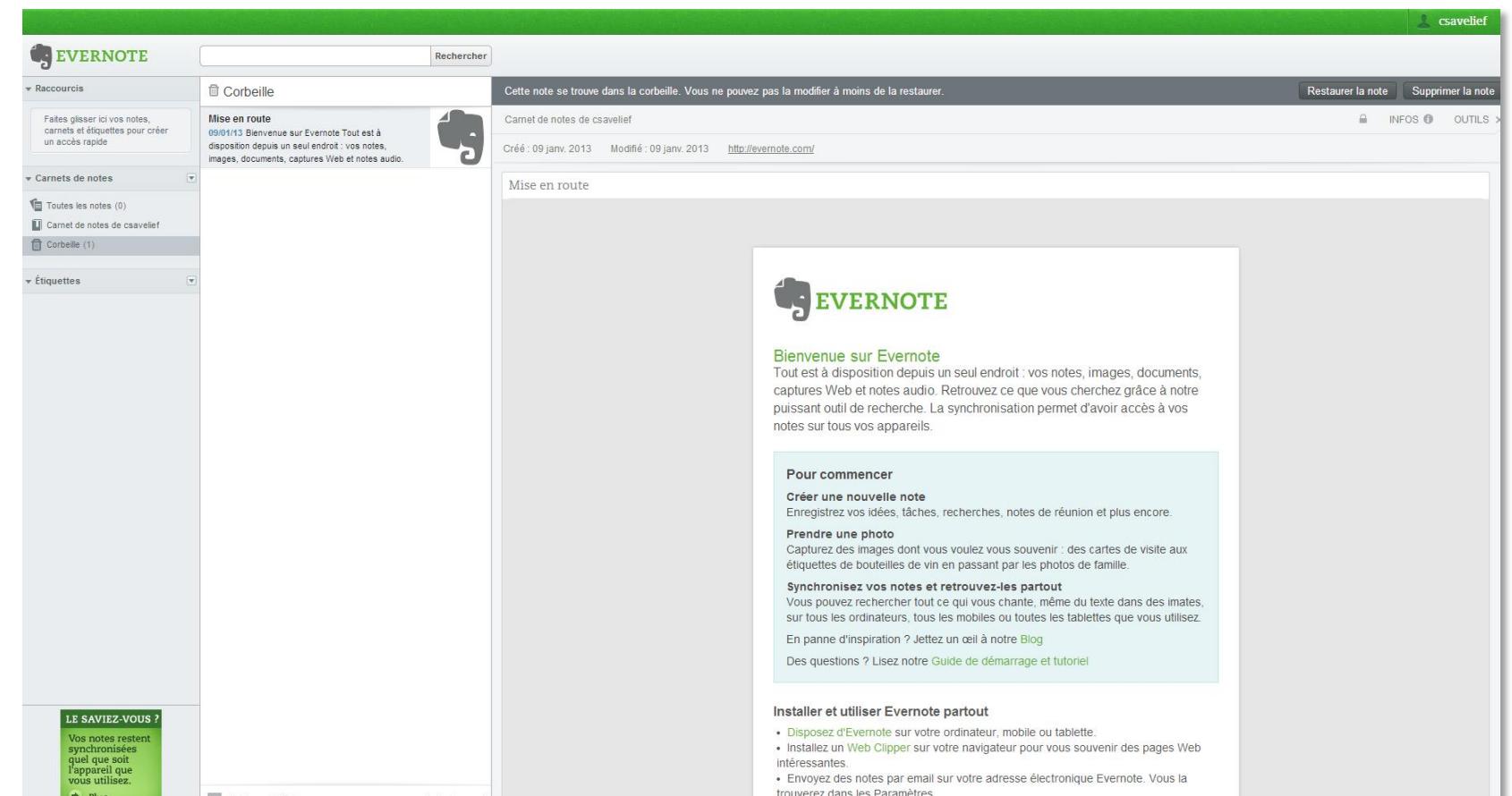
The screenshot shows the homepage of the 'conceptstore' website. It features a banner with two Twitter posts from users 'Zoe' and 'IGN'. Below the banner, there are two sections: 'C'est sûr c'est un vrai "problème"' and 'Illustration des twittos qui ont de vrais problèmes "malheur"'. The footer contains a link to 'http://conceptstore.com'.



The screenshot shows the DoubleClick Ad Exchange interface. On the left, a sidebar lists various status categories: Offers, Open (12 NEW), Negotiations, Awaiting, Waiting, Deals, Active, Ended, Private auc, and Available. The main panel displays a proposal for 'Terms name 1316336205' from 'Web Property 1008' on 'Round 1 of 1'. It includes sections for 'DETAILS', 'Seller', 'Email contacts', 'Point of contact', 'CPM USD 0.1', 'Timeframe 8/9/13 - 8/10/13', 'Advertiser', 'TARGETING', 'Size 728x90', 'Audience segment description N/A', 'TERMS', 'Terms 1316336204', 'MORE INFO', 'Gross estimated spend N/A', 'Estimated impressions per day N/A', and 'COMMENTS N/A'. At the bottom, there are 'Negotiate' and 'Accept' buttons, along with 'Mobile | Desktop' links.



The screenshot shows the Angry Birds online shop landing page. It features a large banner with the text 'ANGRY BIRDS' and images of angry birds and green pigs. Below the banner are sections for 'POPULAR PLUSH TOYS AVAILABLE' and 'ORDER NOW'. At the bottom, there are download links for 'SD', 'HD', 'INSTALL TO PLAY OFFLINE', 'LOGIN', and 'F.A.Q.', along with social sharing links for Twitter and Facebook.



The screenshot shows a detailed view of a note in Evernote. The note is titled 'Corbeille' and contains text about Evernote's synchronization feature. It includes sections for 'Pour commencer', 'Prendre une photo', 'Synchronisez vos notes et retrouvez-les partout', and 'Installer et utiliser Evernote partout'. The footer of the note page includes a link to 'http://evernote.com/'.

# Automation

## G Gradle

Allow you to automate the building, testing and deployment of your app.

- As flexible as Ant
- Dependencies with Maven Central
- Play nicely with Eclipse and IntelliJ



[www.gradle.org](http://www.gradle.org)

# Coding Style

## Why?

- Avoid lengthy arguments
- Facilitate understanding
- Facilitate maintenance
- Make errors obvious

## How?

- Use existing style guides
- Enforce style with tools

**Bonus : one language = one style guide !**

# Question



## How to deal with third-party code?

Use an ADAPTER to convert your interface to their provided interface.

### Benefits:

1. If their interface evolves it will have little to no impact on the rest of your application;
2. It allows you to create boundary tests!

# Tests

## F Fact

“We don’t have enough time to write unit tests.”



# Design by Contract



**What is the difference with testing?**

Testing tries to *diagnose* defects after facts.

Design by Contract tries to *prevent* certain types of defects.

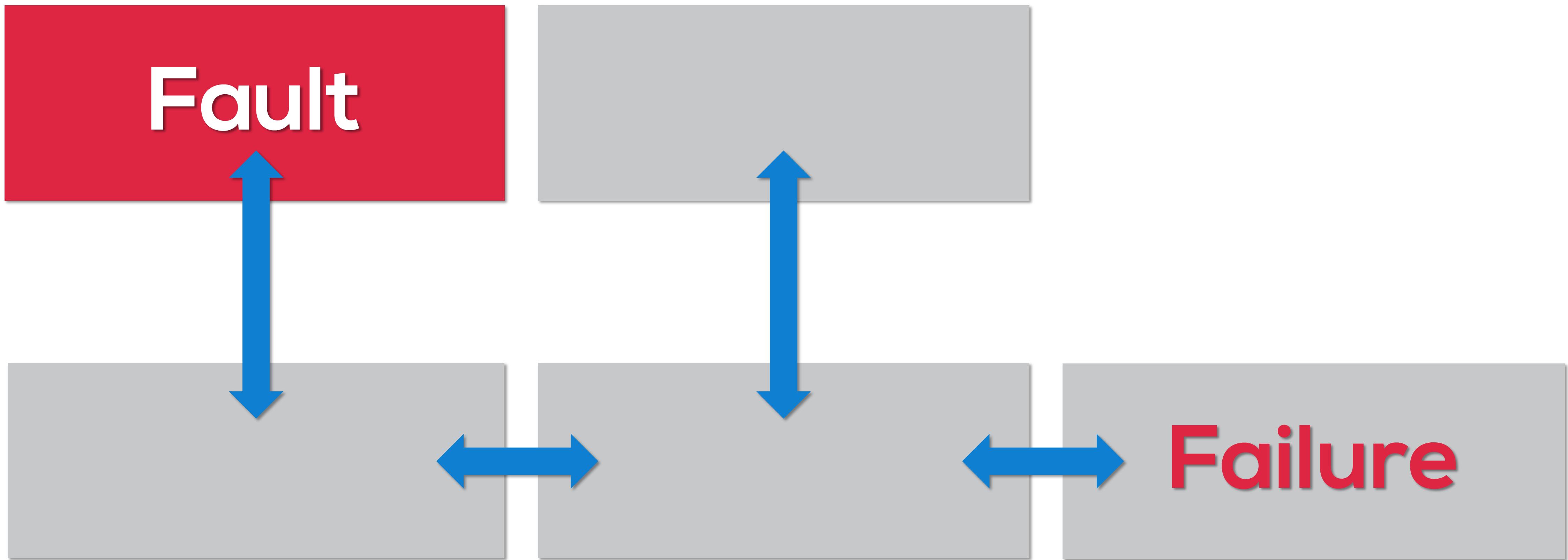


**Design by Contract is particularly useful for...**

...preventing errors in interface between classes.

...preventing errors while reusing classes.

# Design by Contract



# Design by Contract



# Design by Contract

Pre-condition

Post-condition

Invariant

If you promise to call me with the pre-condition satisfied, then I, in return, promise you to deliver a final state in which the post-condition is satisfied.

Furthermore, for all calls you make to me, I will make sure the invariant remains satisfied.

# Example



## Before

```
/**  
 * @param left a sorted list of elements  
 * @param right a sorted list of elements  
 * @return the contents of the two lists, merged, sorted  
 */  
List merge(List left, List right);
```



## After

```
@Requires({  
    "Collections.isSorted(left)",  
    "Collections.isSorted(right)"  
})  
@Ensures({  
    "Collections.containsSame(result, Lists.concatenate(left, right))",  
    "Collections.isSorted(result)"  
})  
List merge(List left, List right);
```



**Contracts for Java**  
<https://code.google.com/p/cofoja/>

# Design by Contract

**Redundant checks:** naive way for including contracts in the source code!

It is the responsibility of the caller to ensure pre-condition.

# Thank You!



## Contact



## Location

1 [dev@mncc.fr](mailto:dev@mncc.fr)

2 [www.mncc.fr](http://www.mncc.fr)

3 [github.com/MNCC](https://github.com/MNCC)

