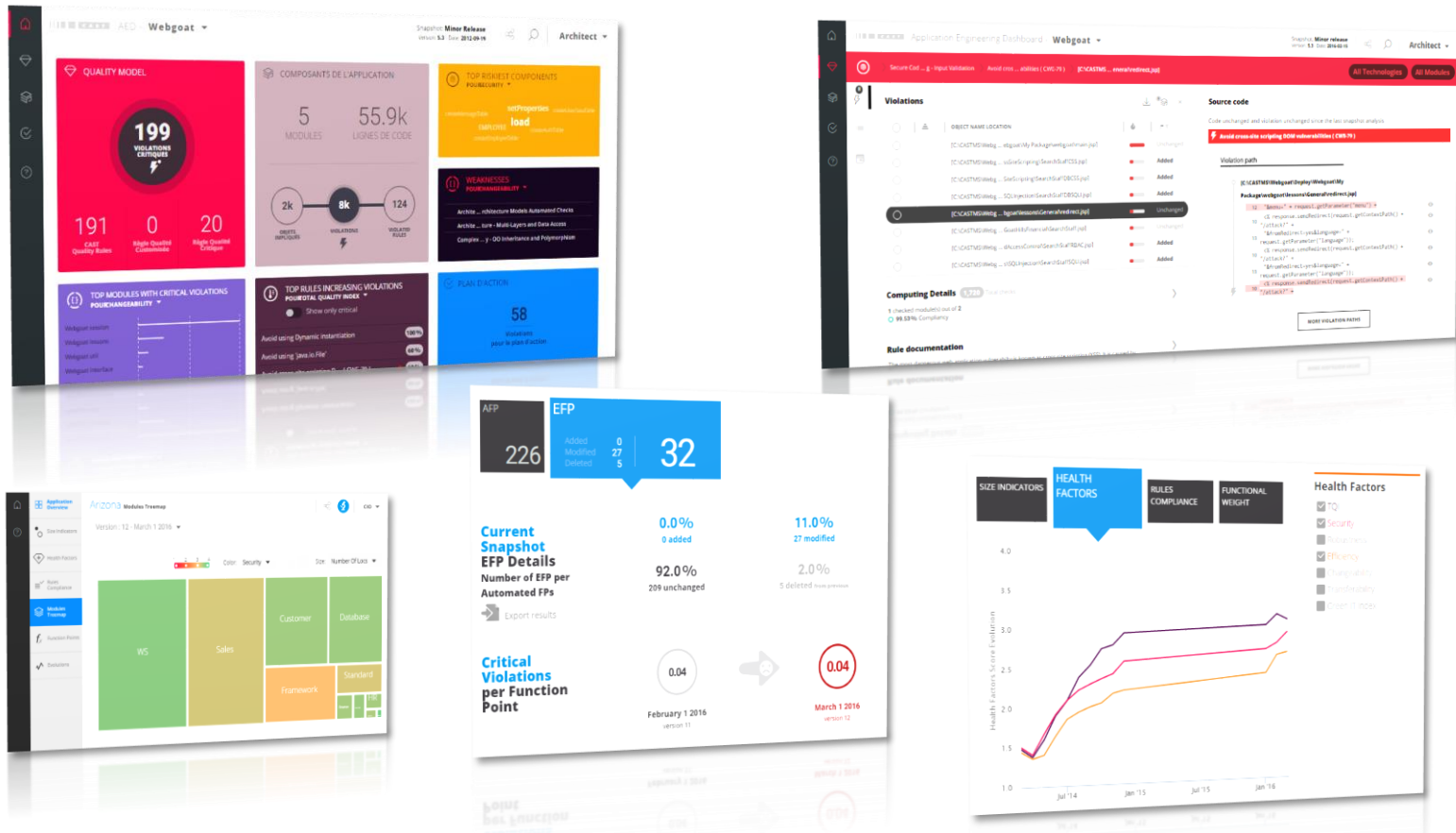


## Un corset pour applications Backbone.js...

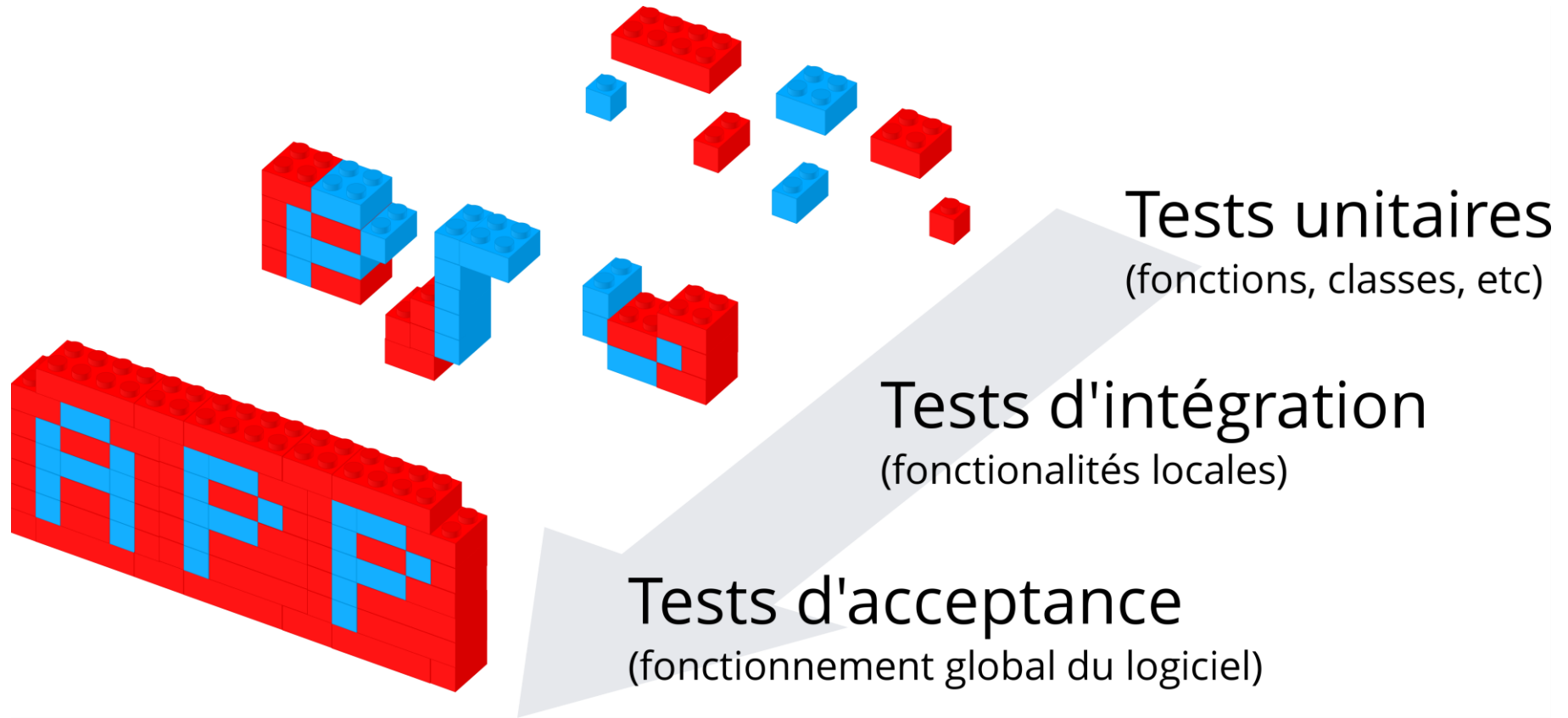
...ou comment les tests unitaires réduisent les scolioses du code

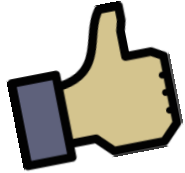


maintenance évolutive – support long terme – criticité métier –



Un bug ou une régression qui atteignent le client ont un gros impact





**Confiance** dans le code

Detection accrue des **regressions**

Prise en compte des **cas limites**

**Documentation** technique du comportement

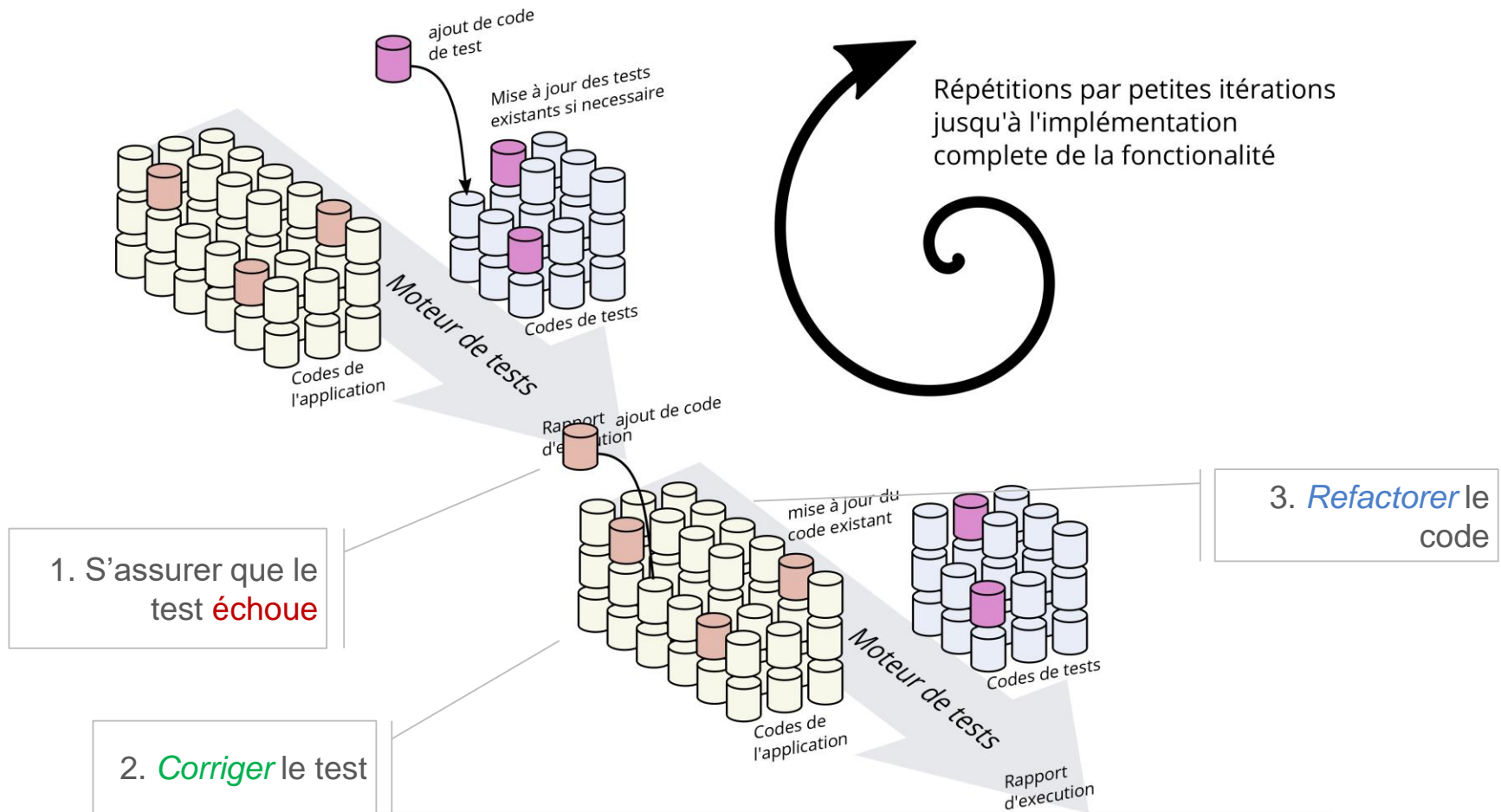
**Couplage faible** du code testé

**Refactoring** serein

...utilisateur heureux...

# Coder le test avant le reste

- Test Driven development (TDD)



# (Quelques) Outils de tests unitaires en JavaScript

Moteurs de tests



Utilitaires



Industrialisation



# Test de modèles

- Avec mocha & chai-expect

```
describe('Test du modèle utilisateurs', function() {  
  
  it('devrait être défini', function() {  
    expect(ModeleUtilisateur).to.be.a('function');  
  });  
  
});
```

Disable automatic test reload

passes: 0 failures: 1 duration: 0.02s 100%

## Composant de gestion des utilisateurs

### Test du modèle utilisateurs

✗ devrait être défini

```
ReferenceError: ModeleUtilisateur is not defined  
    at Context.<anonymous> (http://chaw8lap:8070/CAST-AAD-AED/engineering/test/specs/dashboard/presentation/user.spect.js:  
    at Test.Runnable.run (http://chaw8lap:8070/CAST-AAD-AED/engineering/node_modules/mocha/mocha.js:4336:32)  
    at Runner.runTest (http://chaw8lap:8070/CAST-AAD-AED/engineering/node_modules/mocha/mocha.js:4724:10)  
    at http://chaw8lap:8070/CAST-AAD-AED/engineering/node_modules/mocha/mocha.js:4802:12  
    at next (http://chaw8lap:8070/CAST-AAD-AED/engineering/node_modules/mocha/mocha.js:4649:14)  
    at http://chaw8lap:8070/CAST-AAD-AED/engineering/node_modules/mocha/mocha.js:4659:7  
    at next (http://chaw8lap:8070/CAST-AAD-AED/engineering/node_modules/mocha/mocha.js:4597:23)  
    at http://chaw8lap:8070/CAST-AAD-AED/engineering/node_modules/mocha/mocha.js:4626:5  
    at timeslice (http://chaw8lap:8070/CAST-AAD-AED/engineering/node_modules/mocha/mocha.js:5733:27)
```

# Aller (un peu) plus loin

```
it('doit produire la bonne url REST pour un id donné', function() {
  // setup du test
  var utilisateur = new ModeleUtilisateur({id:123});
  // test et assertions
  expect(utilisateur.url()).to.equal('REST/utilisateurs/123');
});

it('doit retourner le bon age utilisateur', function() {
  var d = new Date('2016-06-22');
  var _Date = Date;
  Date = function(){return d;};
  var utilisateur = new ModeleUtilisateur({id:123,naissance:'1980-06-21'});
  var age = utilisateur.age();
  expect(age).to.equal(36);
  Date = _Date;
});
```

Disable automatic test reload

passes: 2 failures: 1 duration: 0.02s

100%

## Composant de gestion des utilisateurs

### Test du modèle utilisateurs

- ✓ devrait être défini
- ✓ doit produire la bonne url REST pour un id donné
- ✗ doit retourner le bon age utilisateur

AssertionError: expected 35 to equal 36



# Focus sur le test

```
describe('calcul de l\'age', function() {
  var _Date;
  before(function() {
    _Date = Date;
    var d = new Date('2016-06-22');
    Date = function() {return d;};
  });

  after(function() {
    Date = _Date;
  });

  it('doit retourner le bon age utilisateur', function() {
    var utilisateur = new ModeleUtilisateur({id:123,birth:'1980-06-21'});
    expect(utilisateur.age()).to.equal(36);
  });
});
```

Disable automatic test reload

passes: 3 failures: 0 duration: 0.02s 100%

## Composant de gestion des utilisateurs

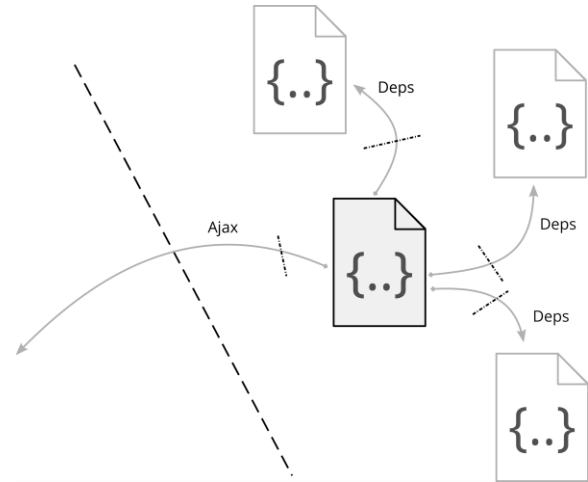
### Test du modèle utilisateurs

- ✓ devrait être défini
- ✓ doit produire la bonne url REST pour un id donné
- ✓ doit retourner le bon age utilisateur



# Cloisonnement des tests

- Avec Sinon.js // **spy, stubs, mocks**
  - Bouchonages, isolation du code en test
  - Simulation de serveurs



```
describe('édition de propriétés', function() {  
  var mockServer;  
  
  beforeEach(function() {  
    mockServer = sinon.fakeServer.create();  
  });  
  
  afterEach(function() {  
    mockServer.restore();  
  });  
  
  ...  
});
```

# Cloisonnement des tests

- Avec Sinon.js // spy, stubs, mocks

```
describe('edition de propriétés', function() {
  var mockServer;

  ...

  it('Save devrait faire un appel put au serveur', function(done) {
    mockServer.respondWith('PUT', 'rest/utilisateurs/123',
      [200, {"Content-Type": "application/json"}, {param: 'une fausse réponse'}]);
    var utilisateur = new ModeleUtilisateur({id:123, birth: '1980-06-21'});
    utilisateur.set({birth: '1980-03-31'});
    utilisateur.save({
      success: function() {
        done();
      },
      fail: function() {
        expect('l\'echec n\'est pas une option').to.be.false;
      }
    });
    mockServer.respond();
  });
});
```

# Test de vues

- Quoi tester ?
  - Affichage correct de valeurs  
// par ex. formatages de nombres
  - Appels d'évenements sur actions
- Quoi ne pas tester ?
  - Éléments susceptibles de changer (fréquemment)
  - Structures techniques // par ex. DOM

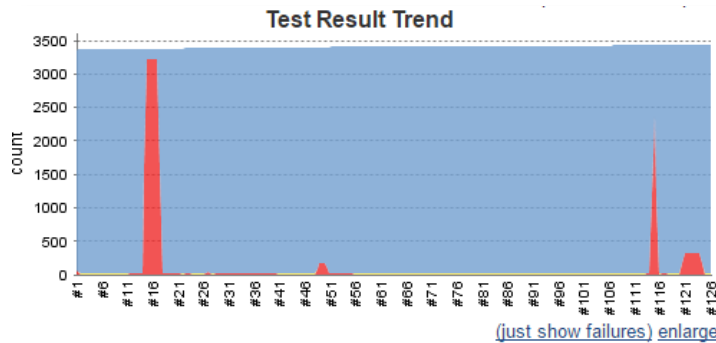
```
describe('Vue utilisateur', function() {  
  var vue;  
  beforeEach(function() {  
    var user = new ModeleUtilisateur({name: 'Alice', sex: 'femme'});  
    vue = new VueUtilisateur({model: user});  
    vue.render();  
  });  
  
  it('verifier le contenu', function() {  
    var nom = vue.$el.find('h1').html();  
    expect(nom).toEqual('Alice');  
  });  
  
  it('test d\'evenement', function() {  
    var eventMethodStub = sinon.stub(vue, 'edit');  
    vue.$el.find('.edition').click();  
    expect(eventMethodStub.calledOnce).toBe(true);  
  });  
});
```

# Test de routes

```
describe("tester les routes", function() {  
    var router, routeSpy;  
  
    beforeEach(function() {  
        router = new AppRouter();  
        routeSpy = sinon.spy();  
        try {  
            Backbone.history.start({silent:true, pushState:true});  
        } catch(e) {}  
        router.navigate("elsewhere"); // default location  
    });  
  
    it('url vide vers la page d\'index', function() {  
        router.bind('route:index', routeSpy);  
        router.navigate('', true);  
        expect(routeSpy.calledOnce).to.be.ok;  
    });  
});
```

# Aller plus loin

- Pilotage des tests sur *vraies* navigateurs // karma, browser-stack, etc.
- Couverture du code par les tests // karma coverage
- Tests d'intégration et d'acceptance // n'oublions pas la perf !!



Project Coverage summary

Name	Packages	Files	Classes	Methods	Lines	Conditionals
Cobertura Coverage Report	100% 5/5	100% 5/5	100% 5/5	13% 63/474	14% 406/2811	9% 142/1533

## Test Result

0 failures (±0)

339 tests (+3)  
Took 0.21 sec.

## All Tests

Package	Duration	Fail	(diff) Skip	(diff) Pass	(diff) Total	(diff)
<a href="#">Chrome 51.0.2704 (Windows 7)</a>	38 ms	0	0	113 +1	113 +1	
<a href="#">Firefox 47.0.0 (Windows 7)</a>	80 ms	0	0	113 +1	113 +1	
<a href="#">IE 11.0.0 (Windows 7)</a>	90 ms	0	0	113 +1	113 +1	

```
235     },
236     list: function() {
237 3       var application, _i, _len, _ref;
238 3       if (this.adgValidModels) {
239 1         return this.adgValidModels.slice(0);
240     }
241 2     this.adgValidModels = [];
242 2     _ref = this.models;
243 2     for (_i = 0, _len = _ref.length; _i < _len; _i++) {
244 4       application = _ref[_i];
245 4       if (application.getId() != null) {
246 4         this.adgValidModels.push(application);
247     }
248     }
249 2     return this.adgValidModels.slice(0);
250   }
}
```