Validation continue des exigences et de l'implémentation

méthode et techniques

Mathieu Acher

Maître de Conférences

mathieu.acher@irisa.fr

UNIVERSITÉ DE RENNES 1

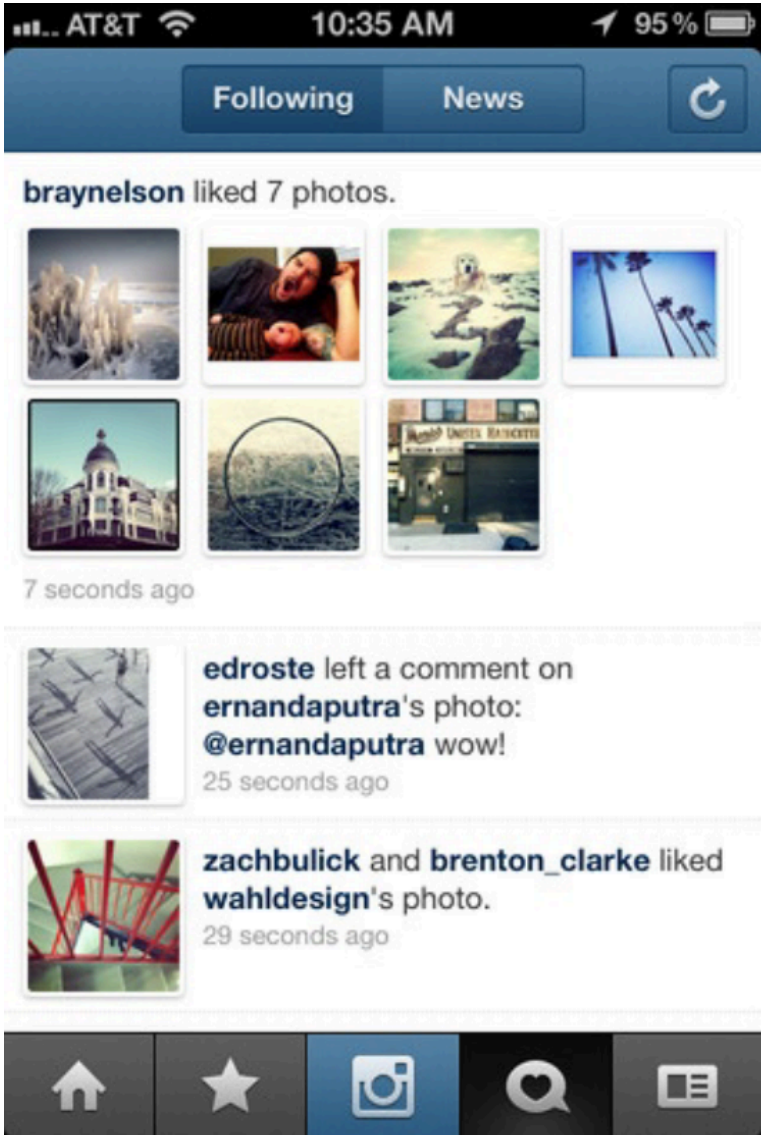
istic Informatique Électronique

Material

http://mathieuacher.com/teaching/PDL/

Deux projets:
Un succès
Un échec
Des défils

Instagram Story



Instagram Story

« Instagram is an app that **only took 8 weeks** to build and ship, but was a product of over a year of work. »

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

« While I was there working in marketing, I started doing more and more engineering at night on simple ideas that helped me learn how to program **(I don’t have any formal CS degree or training) »**

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

« We spent 1 week prototyping a version that focused solely on photos.
It was pretty awful. So we went back to creating a native version of Burbn. We actually got an entire version of Burbn done as an iPhone app, but it felt cluttered, and overrun with features. It was really difficult to decide to start from scratch, but we went out on a limb, and basically cut everything in the Burbn app except for its photo, comment, and like capabilities. What remained was Instagram. »

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

« So 8 weeks later, we gave it to our friends, beta tested, bug fixed, etc. and this Monday we decided it was ready to ship. »

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

« Who is responsible for Instagram's UI design?

For better or for worse, I've done most of the pixel pushing in our app. ;) »

http://www.forbes.com/sites/limyunghui/2012/04/09/inspiring-insights-by-instagram-ceo-kevin-systrom-the-man-who-built-a-1-billion-startup/

Instagram Story

- 30+ millions d'utilisateur en 2 ans
- 25k inscriptions le premier jour
  - « best & worst day of our lives so far »
  - « favicon » cause des milliers d'erreurs 404
    - « 404-ing on Django, causing tons of errors »
- Un seul serveur au lancement
  - Moins puissant qu'un MacBook Pro
- La suite: passage à l'échelle, cloud (EC2) et ingénierie du logiciel

https://speakerdeck.com/mikeyk/scaling-instagram
http://zoompf.com/blog/2012/04/instagram-and-optimizing-favicons

Instagram Story

- Sur la trentaine de composants, 4 seulement ont été écrits à partir de zéro
  - App iOS, App Android, Android Push Notification Service et Redis Query analyzer

node2dm

Fabric

Instagram Story (key lessons)

- Sélection et intégration de multiples librairies

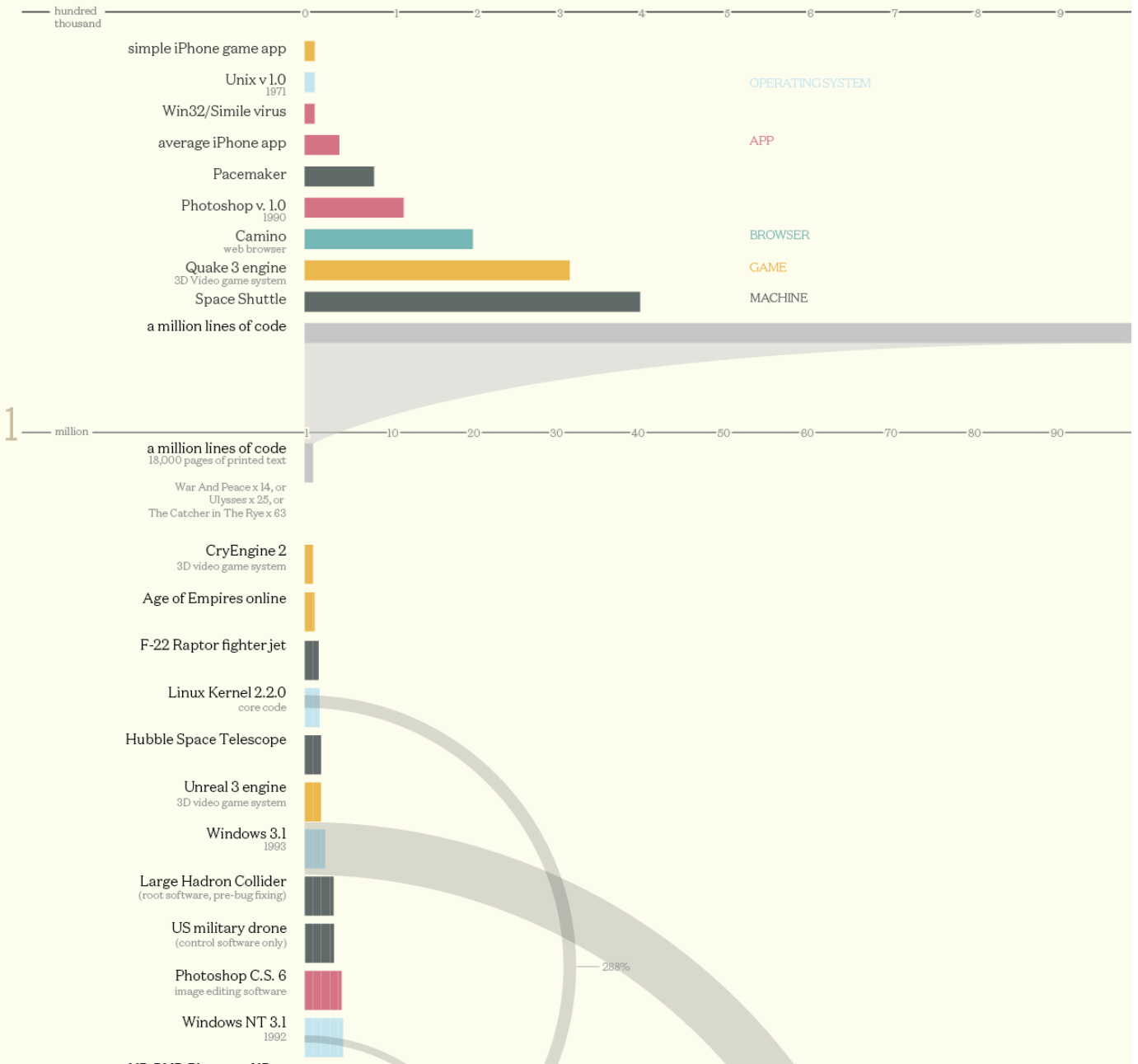
- Open source community
  - Apprendre, partager, demander, répondre, etc.
- Auto-apprentissage
  - « Product guys » sont maintenant à même de rivaliser…

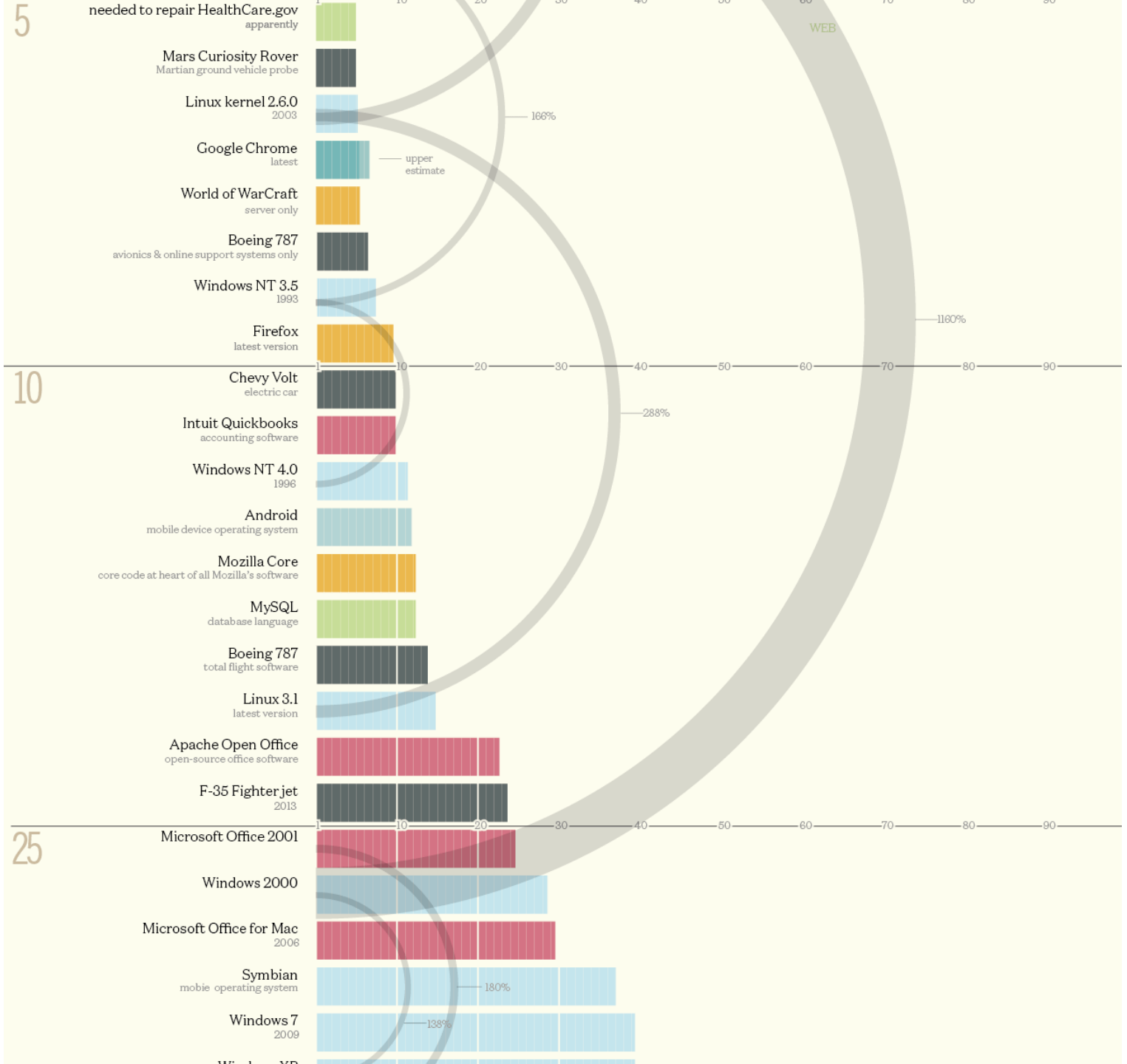
- Agilité, développement incrémental

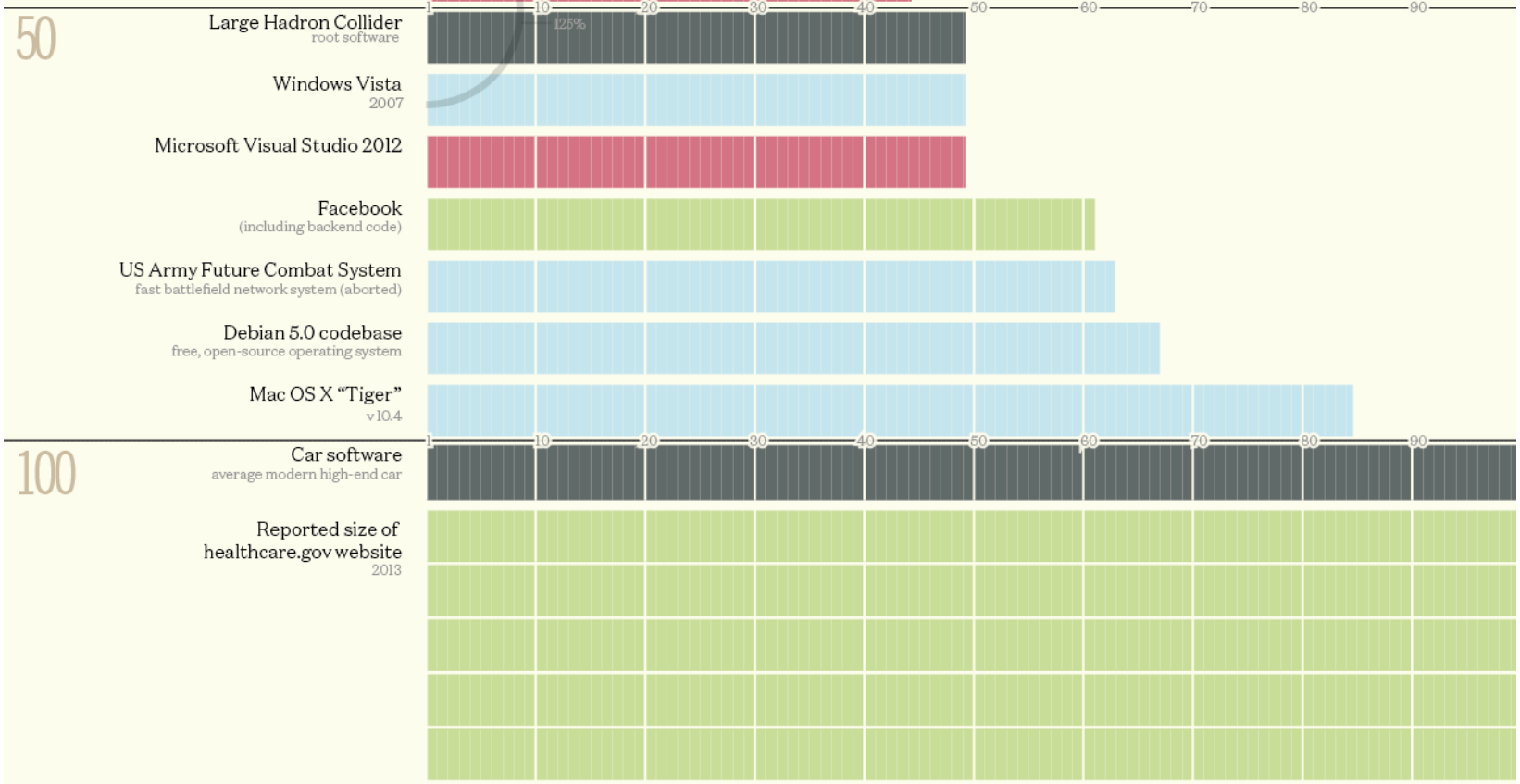
http://instagram-engineering.tumblr.com/

Codebases

Millions of lines of code




5
needed to repair HealthCare.gov
apparently
Mars Curiosity Rover
Martian ground vehicle probe
Linux kernel 2.6.0
2003
Google Chrome
latest
upper estimate
World of WarCraft
server only
Boeing 787
avionics & online support systems only
Windows NT 3.5
1993
Firefox
latest version
10
Chevy Volt
electric car
Intuit Quickbooks
accounting software
Windows NT 4.0
1996
Android
mobile device operating system
Mozilla Core
core code at heart of all Mozilla's software
MySQL
database language
Boeing 787
total flight software
Linux 3.1
latest version
Apache Open Office
open-source office software
F-35 Fighter jet
2013
25
Microsoft Office 2001
Windows 2000
Microsoft Office for Mac
2008
Symbian
mobie operating system
Windows 7
2009
1 10 20 30 40 50 60 70 80 90
WEB
166%
1160%
288%
180%
138%



concept & design: David McCandless

informationisbeautiful.net

research: Pearl Doughty-White, Miriam Quick

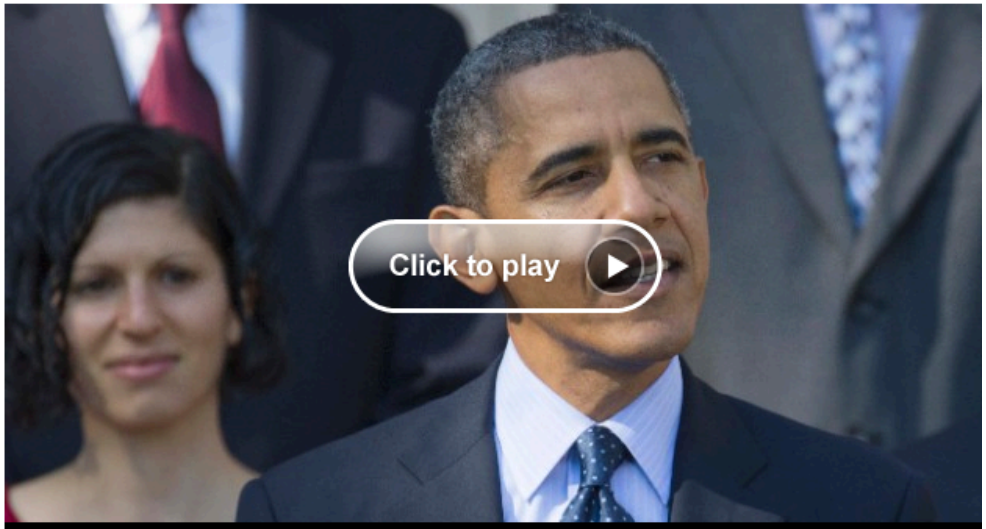
work in progress

v 0.62 // Oct 2013

sources NASA, Quora, Ohloh, Wired & press reports
note some guess work, rumours & estimates
data bit.ly/KIB_linescode

Report: Healthcare website failed test ahead of rollout

By **Ed Payne, Matt Smith** and **Tom Cohen**, CNN
October 23, 2013 -- Updated 0103 GMT (0903 HKT)



Report: Obamacare site failed early test

STORY HIGHLIGHTS

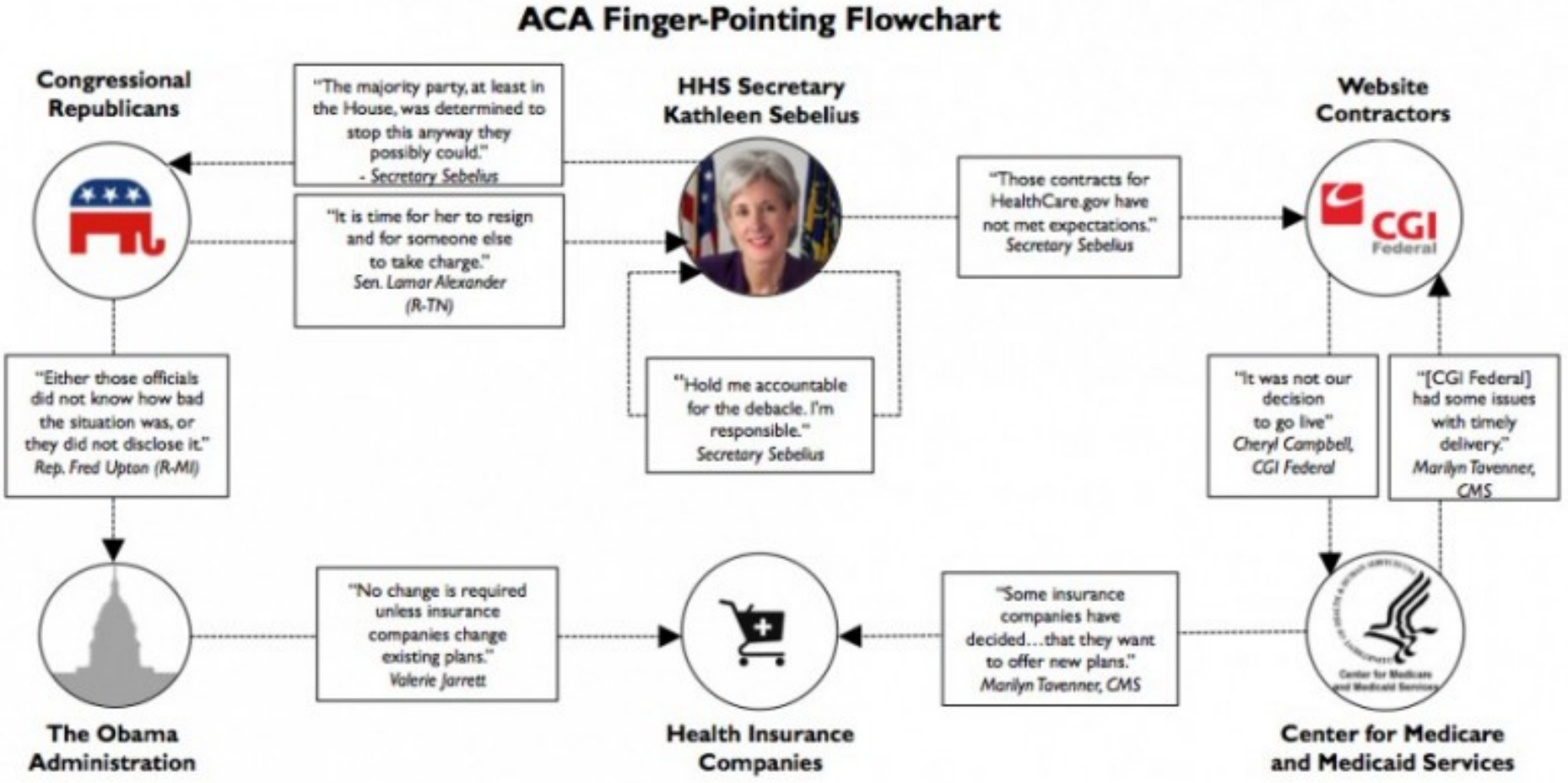
- **NEW:** Top White House official part of "tech surge" on Obamacare
- Obamacare "is not failing" despite website woes, White House spokesman says
- Obama says HealthCare.gov problems are "going to get fixed"
- Secretary Sebelius expected to testify at a congressional hearing next week

Washington (CNN) -- The President's healthcare sign-up web page was supposed to handle tens of thousands of people at once. But in a trial run days before its launch, just a few hundred users flatlined the site.

Despite the problems, federal health officials pushed aside the crash cart and rolled out HealthCare.gov on October 1 as planned, The Washington Post reported.

The result? The website crashed shortly after midnight as a couple thousand people tried to start the process, two people familiar with the project told the Post.

Requirements engineering/ Management problem



http://www.washingtonpost.com/blogs/wonkblog/wp/2013/11/01/thirty-one-things-we-learned-in-healthcare-govs-first-31-days/

Thirty-one things we learned in HealthCare.gov's first 31 days

Scalability problem

Technical problems (e.g., inaccurate data, cancellation failures)

Testing issues

http://www.washingtonpost.com/blogs/wonkblog/wp/2013/11/01/thirty-one-things-we-learned-in-healthcare-govs-first-31-days/

10. HealthCare.gov didn't have enough testing before going live.

This became clear in a series of Congressional hearings, where federal contractors testified that end-to-end testing only began in the final weeks of September, right before the Oct. 1 launch. When pressed on how much time would have been ideal for testing, one contractor told lawmakers that "months would have been nice."

http://www.washingtonpost.com/blogs/wonkblog/wp/2013/11/01/thirty-one-things-we-learned-in-healthcare-govs-first-31-days/

1 succès, 1 échec

- 1 succès:
  - réutilisation: sélection et intégration de multiples librairies
  - agilité, développement incrémental: les exigences ne sont pas fixes; sorties d'un produit qui correspond aux attentes des utilisateurs
- 1 échec:
  - problèmes dans la communication et l'élicitation des exigences
  - pas de test

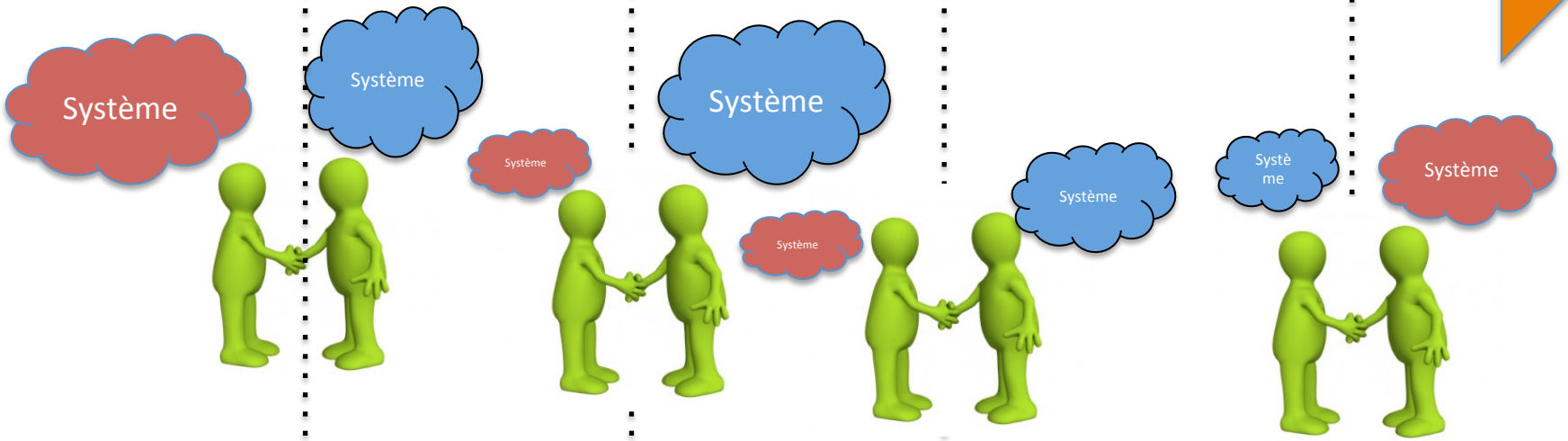
Votre projet = succès + !échec

Votre projet

- Réutilisation: sélection et intégration de multiples librairies
- Agilité, développement incrémental: les exigences ne sont pas fixes; sorties d'un produit qui correspond aux attentes des utilisateurs
- Communication et élicitation des exigences avec le client; modélisation
- Test

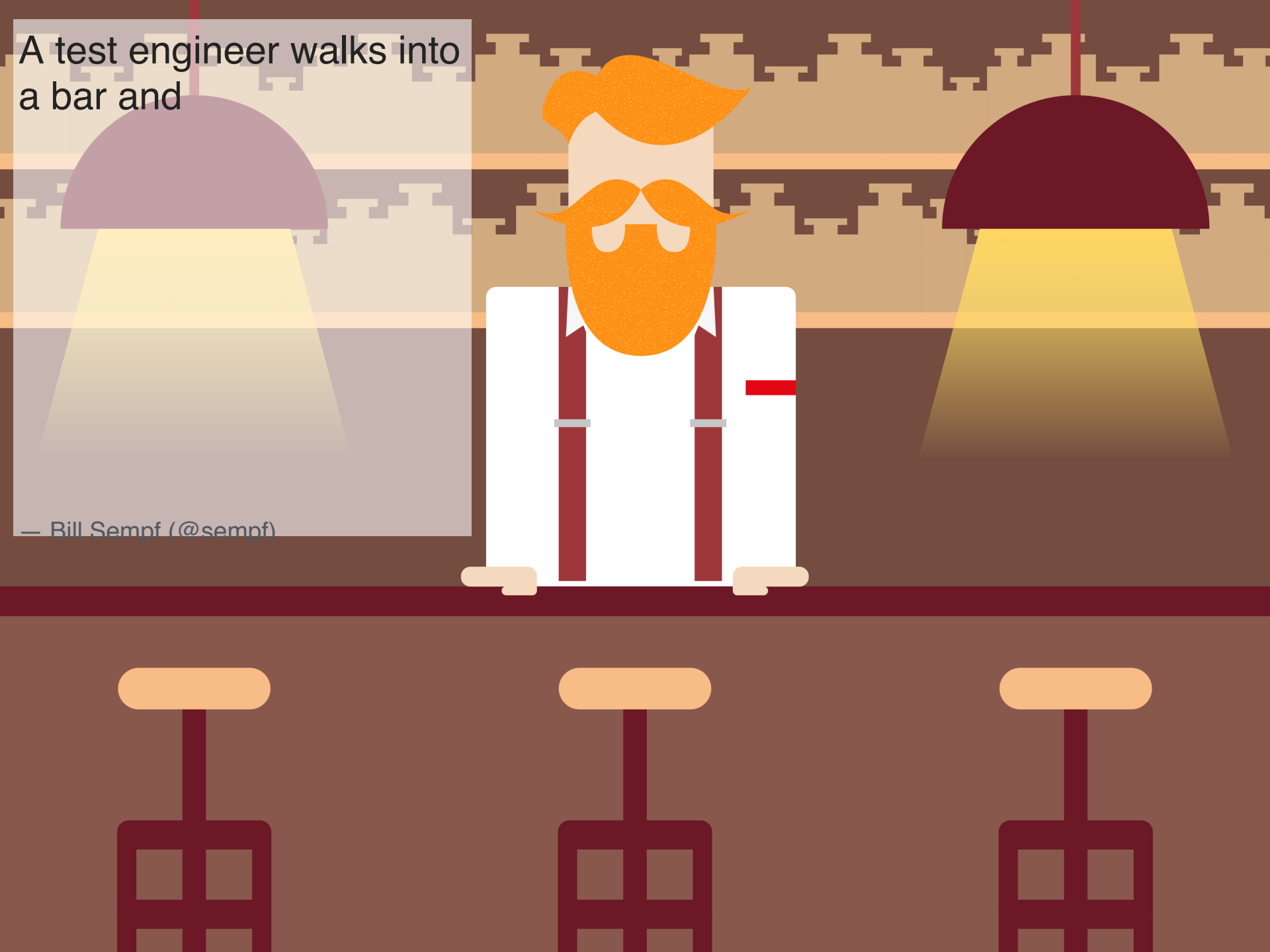
EX (exigences; cahier des charges)

SP (sprints; implémentation)



Valider à chaque itération avec le client: montrer les exigences et l'implémentation (le « produit » en action)

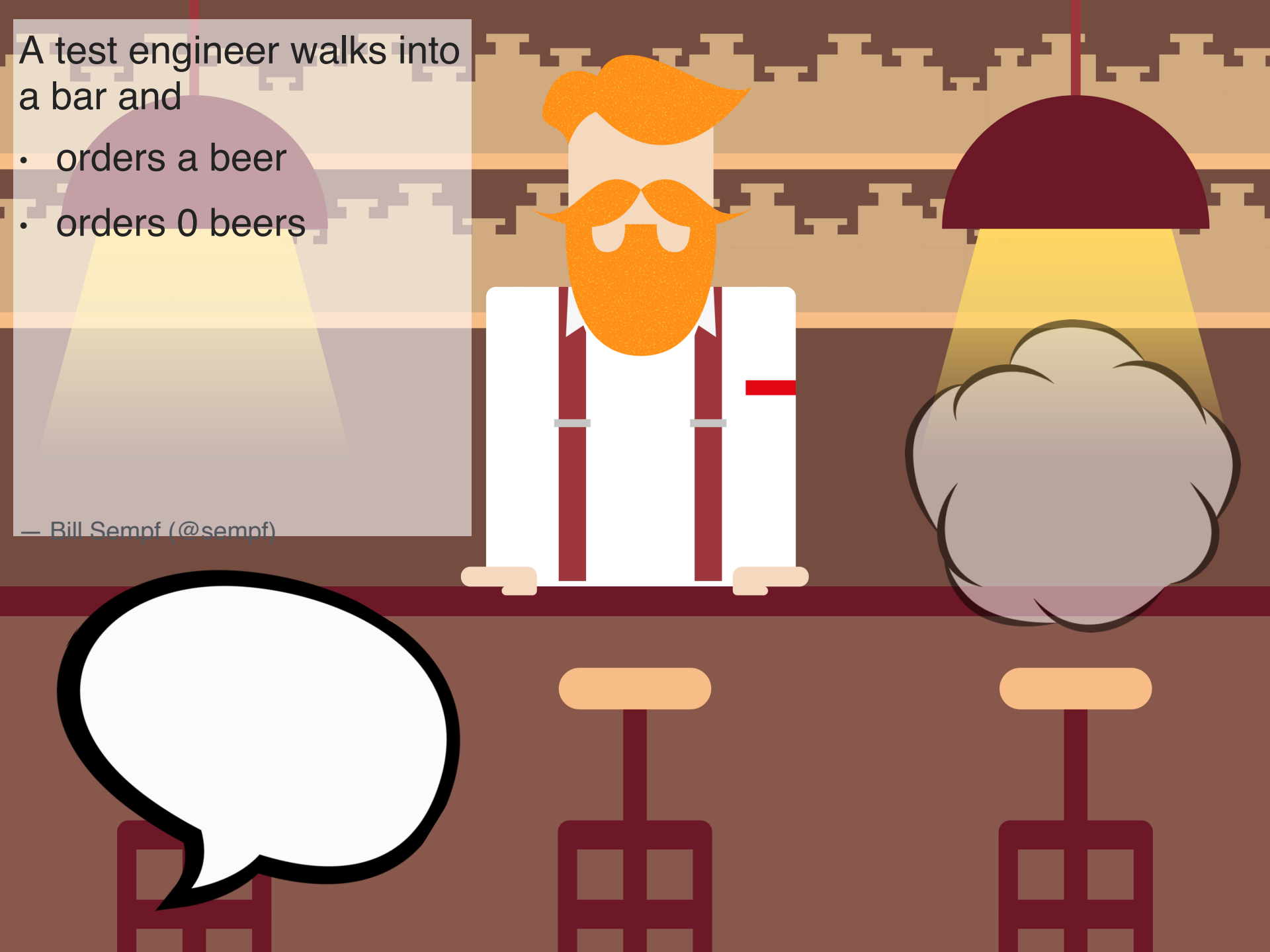
Test


A test engineer walks into a bar and
— Bill Sempf (@sempf)

A test engineer walks into a bar and

- orders a beer

— Bill Sempf (@sempf)

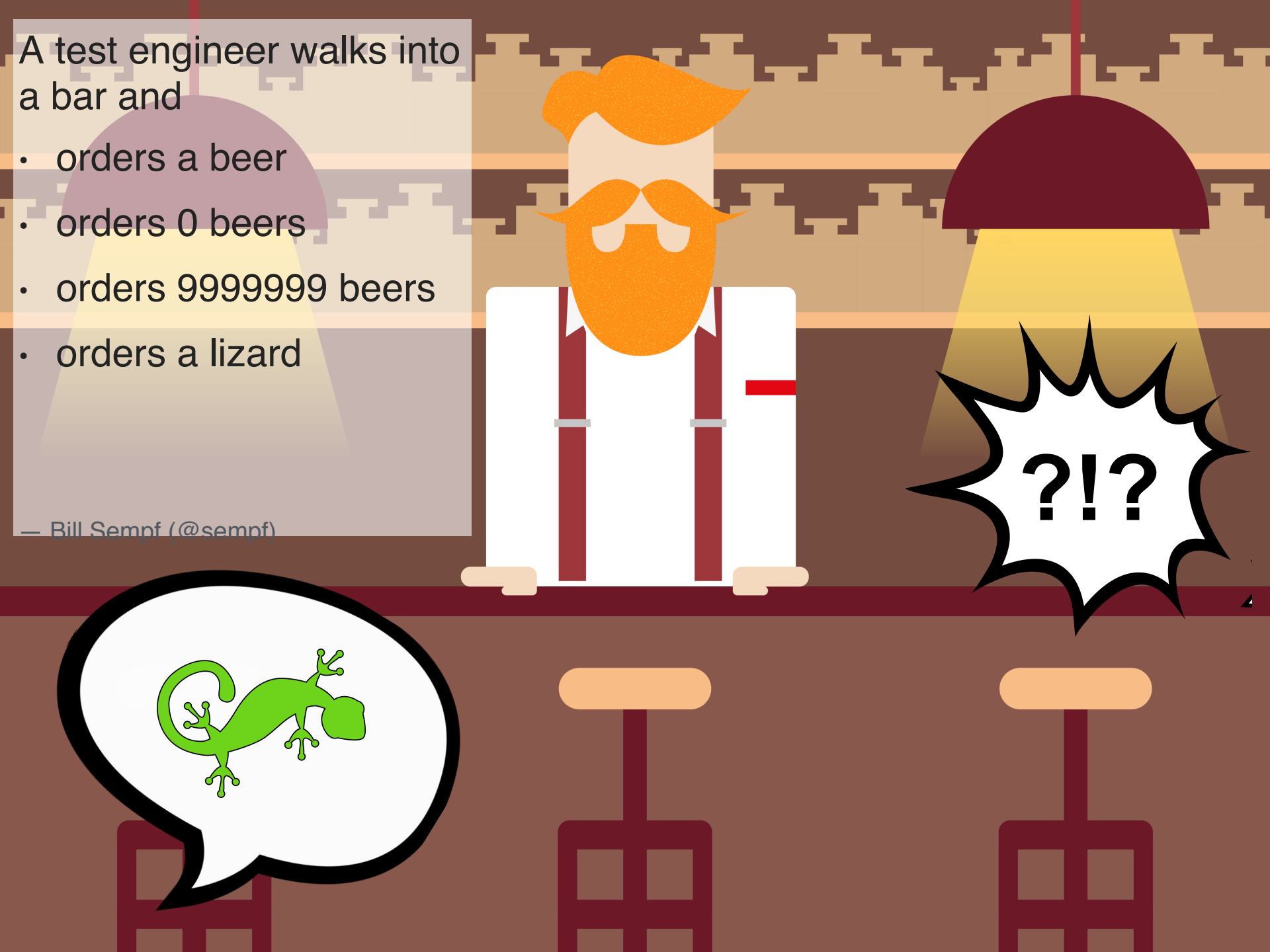

A test engineer walks into a bar and
• orders a beer
• orders 0 beers
— Bill Sempf (@sempf)


A test engineer walks into a bar and
• orders a beer
• orders 0 beers
• orders 9999999 beers
— Bill Sempf (@sempf)
9,999,999

A test engineer walks into a bar and

- orders a beer
- orders 0 beers
- orders 9999999 beers
- orders a lizard

— Bill Sempf (@sempf)

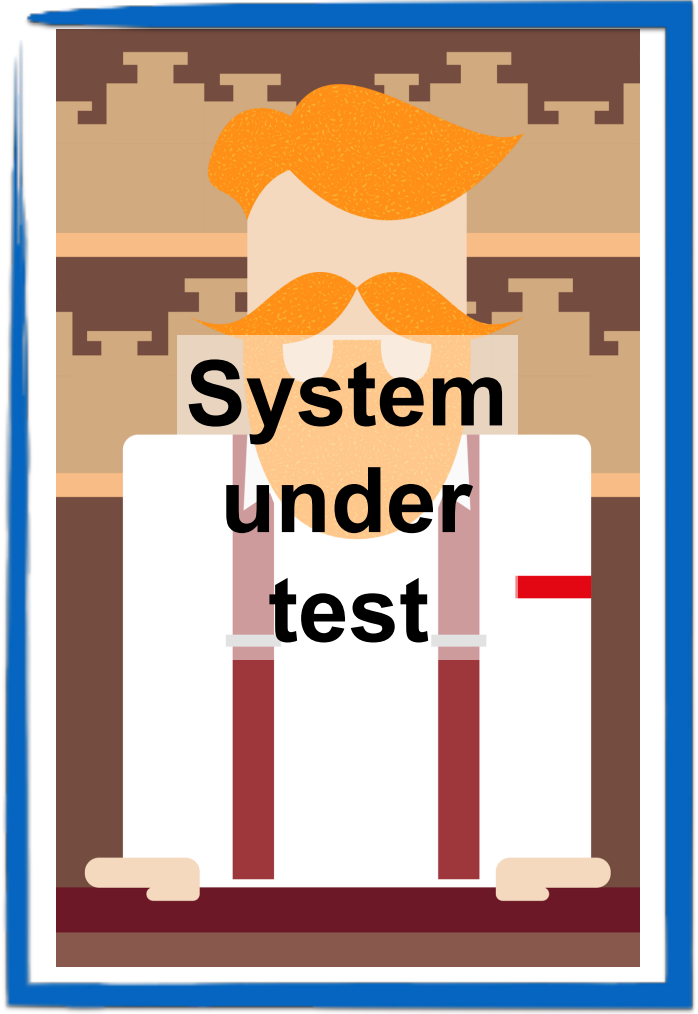


A test engineer walks into a bar and

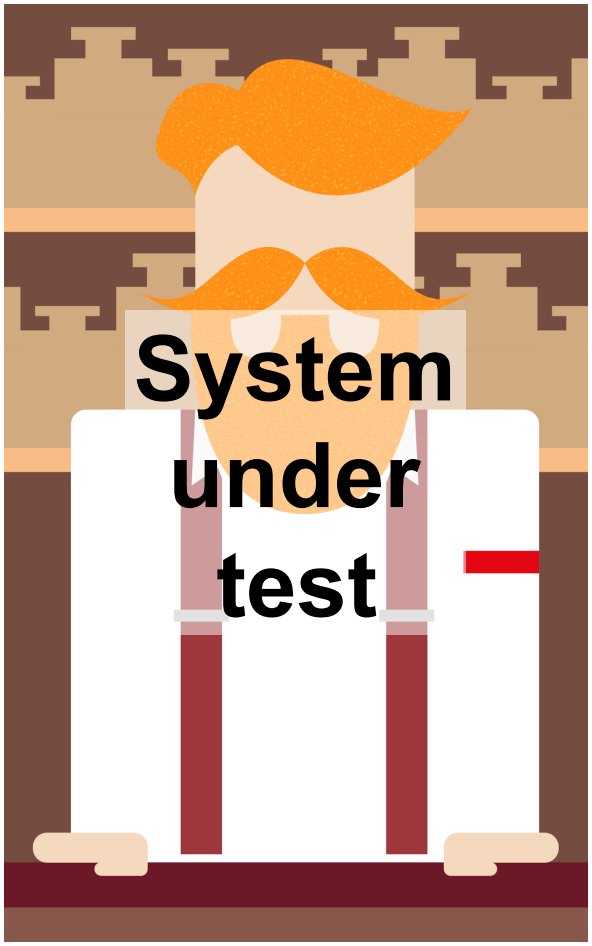
- orders a beer
- orders 0 beers
- orders 9999999 beers
- orders a lizard
- orders -1 beers

— Bill Sempf (@sempf)

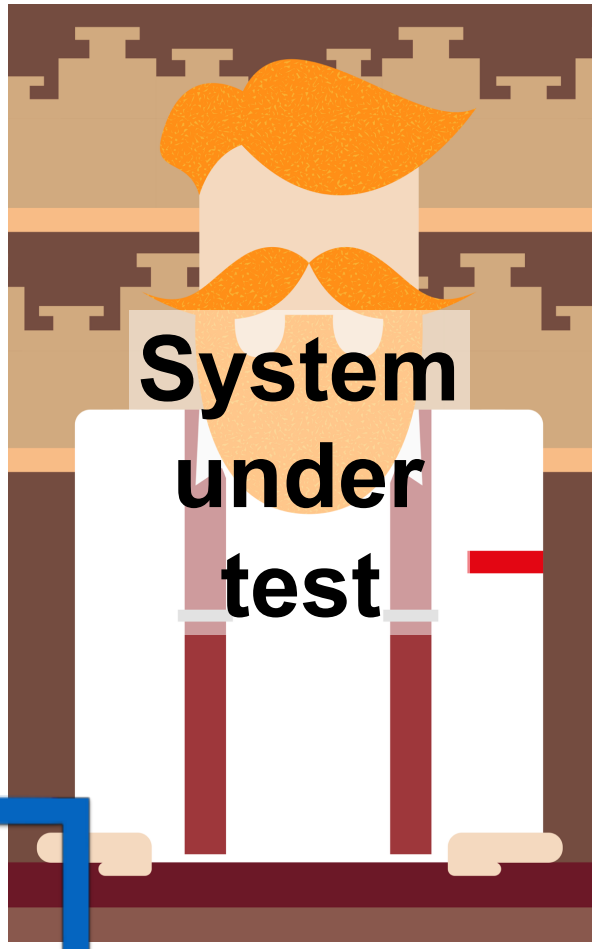

A test engineer walks into a bar and
• orders a beer
• orders 0 beers
• orders 9999999 beers
• orders a lizard
• orders -1 beers
• orders a "sfdeljknesv"
— Bill Sempf (@sempf)
?!?
"sfdeljknesv"

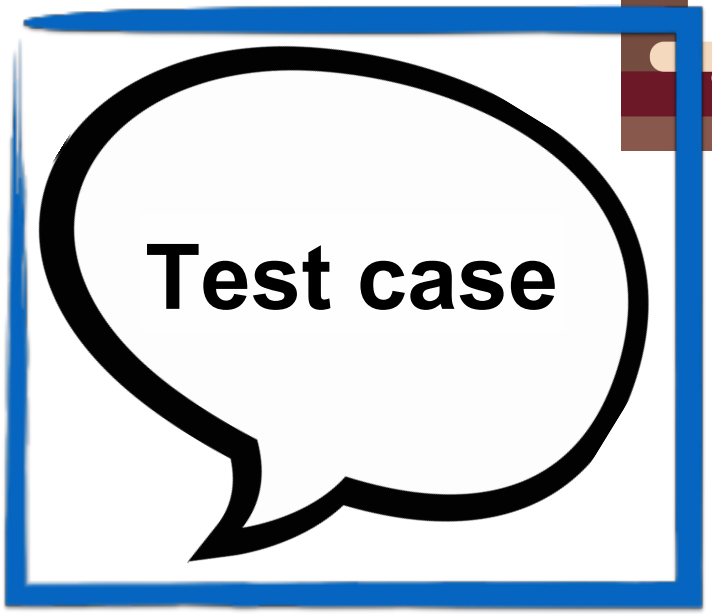

System
under
test

Specification

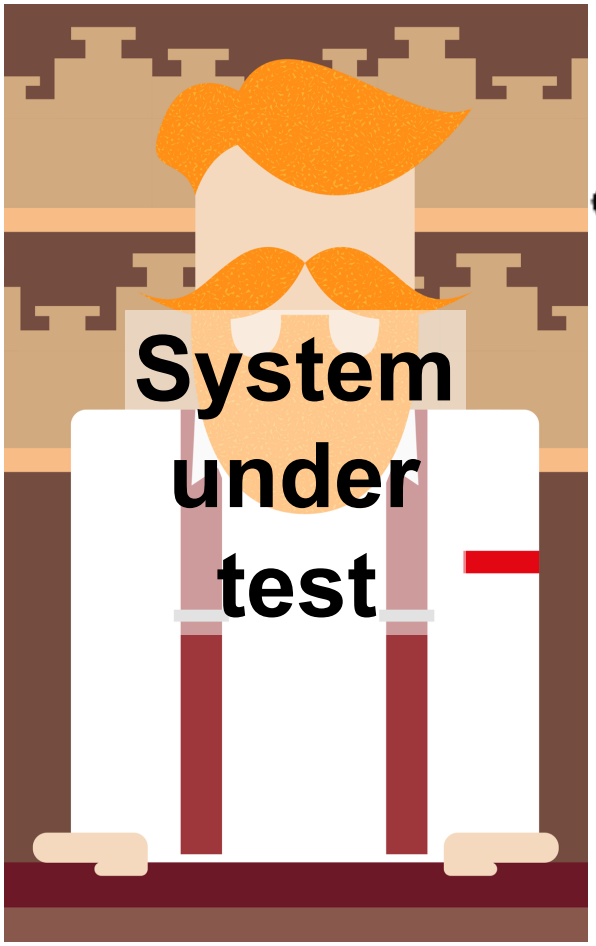


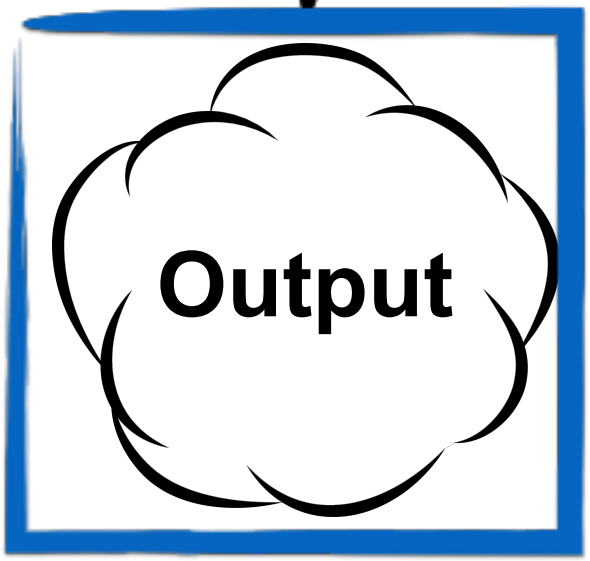
Specification



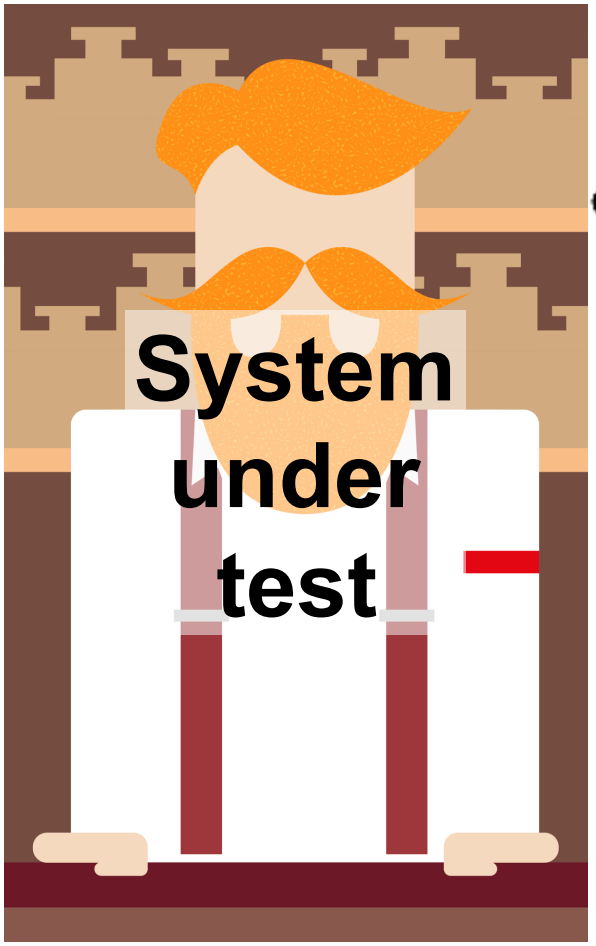


Specification





Specification



Output

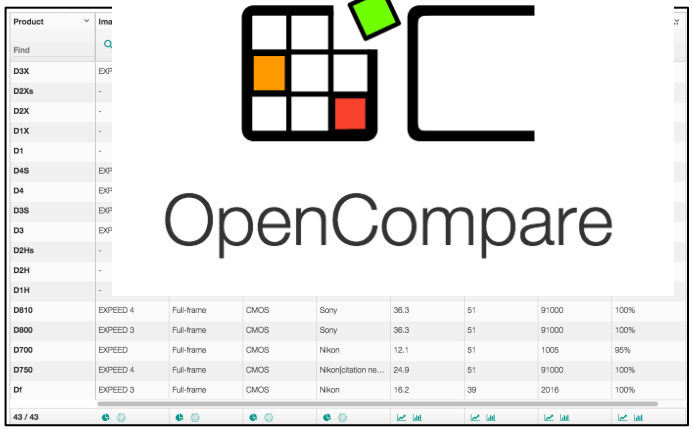
Test case

Pass or fail?

Tests et projets

Projet #1

OpenCompareReverseJSON



OpenCompare

- Java API (incl. JSON parsing)
- JSON Schema
- UML Class diagram

Projet #2

OpenFoodFacts2CSV

OPEN FOOD FACTS
l'information alimentaire ouverte

CSV
(Comma Separated Values)

Product	Image process.	Sensor format	Sensor type	Sensor manufa.	Megapixels	Focus points	Metering pixels	Viewfinder cov.
Find								
D3X	EXPEED	Full-frame	CMOS	Sony	24.5	51	1005	100%
D2Xs	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D2X	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D1X	-	APS-C	CCD	Sony	5.3	5	1005	96%
D1	-	APS-C	CCD	Sony	2.66	5	1005	96%
D4S	EXPEED 4	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D4	EXPEED 3	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D3S	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D3	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D2Hs	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D2H	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D1H	-	APS-C	CCD	Sony	2.7	5	1005	96%
D810	EXPEED 4	Full-frame	CMOS	Sony	36.3	51	91000	100%
D800	EXPEED 3	Full-frame	CMOS	Sony	36.3	51	91000	100%
D700	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	96%
D750	EXPEED 4	Full-frame	CMOS	Nikon[citation ne...	24.9	51	91000	100%
Df	EXPEED 3	Full-frame	CMOS	Nikon	16.2	39	2016	100%
43 / 43								

OpenCompare

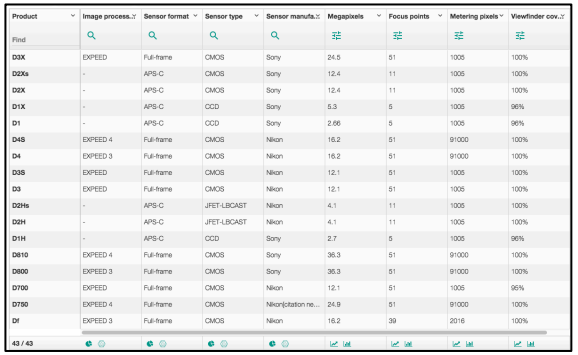
Projet #3

MatrixSynthesizerWikipedia

WIKIDATA

WIKIPEDIA
The Free Encyclopedia

CSV
(Comma Separated Values)



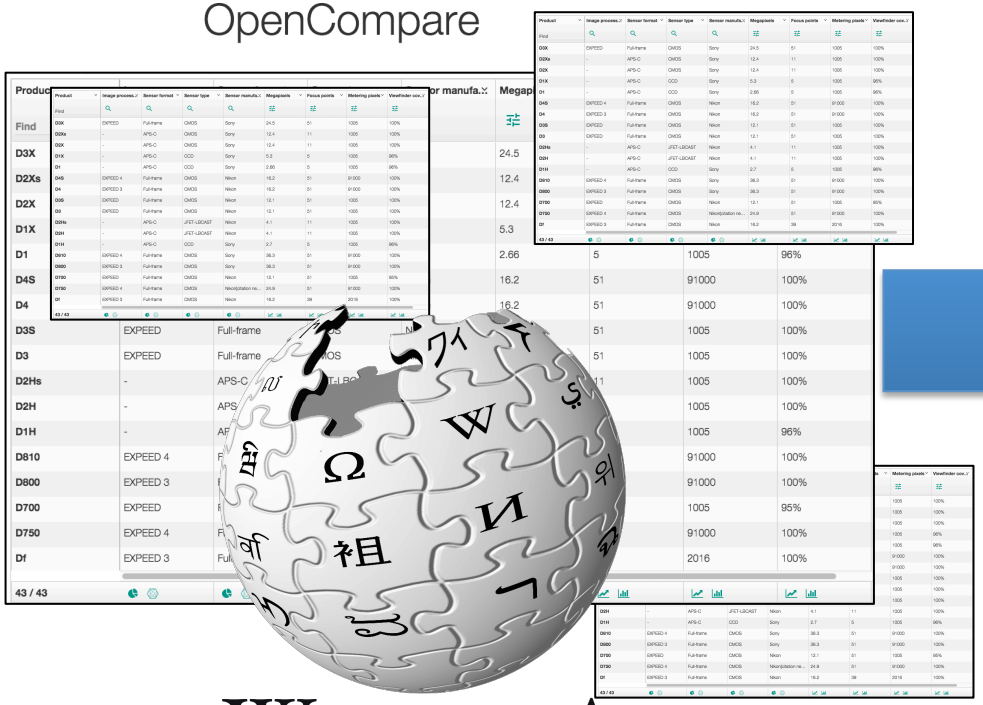
Product	Image process..	Sensor format	Sensor type	Sensor manufa..	Megapixels	Focus points	Metering pixels	Viewfinder cov..
Find								
D3X	EXPEED	Full-frame	CMOS	Sony	24.5	51	1005	100%
D2Xs	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D2X	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D1X	-	APS-C	CCD	Sony	5.3	5	1005	96%
D1	-	APS-C	CCD	Sony	2.66	5	1005	96%
D4S	EXPEED 4	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D4	EXPEED 3	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D3S	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D3	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D2Hs	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D2H	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D1H	-	APS-C	CCD	Sony	2.7	5	1005	96%
D810	EXPEED 4	Full-frame	CMOS	Sony	36.3	51	91000	100%
D800	EXPEED 3	Full-frame	CMOS	Sony	36.3	51	91000	100%
D700	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	95%
D750	EXPEED 4	Full-frame	CMOS	Nikon[citation ne...	24.9	51	91000	100%
Df	EXPEED 3	Full-frame	CMOS	Nikon	16.2	39	2016	100%
43 / 43								

OpenCompare

Projet #4

WikipediaMatrixAnalysis

OpenCompare



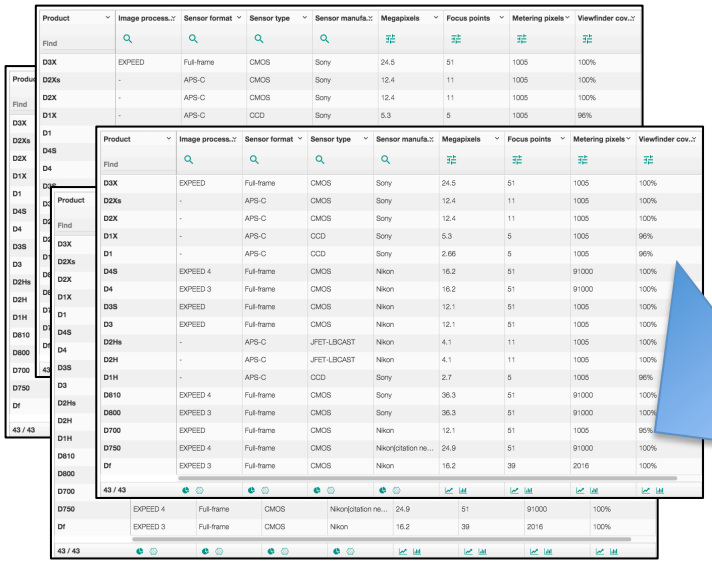
WIKIPEDIA
The Free Encyclopedia

Automated
Analysis

4 Projets et des risques

- Activités similaires:
  1. Eliciter et valider des exigences
  2. Développement Java pour traiter des données
  3. Travail collaboratif
- **Risques** similaires. Exemples:
  1. Implémenter des fonctionnalités inutiles ou qui ne correspondent pas aux besoins du client
  2. Le programme Java est incapable de traiter certains types de données
  3. La sortie générée (e.g., CSV) n'est pas "correcte"
  4. Une modification dans 1, 2, et 3 ne permet pas à un membre du groupe de correctement contribuer

Comment fait-on ?



Product	Image process.	Sensor format	Sensor type	Sensor manufa.	Megapixels	Focus points	Metering pixels	Viewfinder cov.
D3X	EXPEED	Full-frame	CMOS	Sony	24.5	51	1005	100%
D2Xs	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D2X	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D1X	-	APS-C	CCD	Sony	5.3	5	1005	96%
D1	-	APS-C	CCD	Sony	2.66	5	1005	96%
D4S	EXPEED 4	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D4	EXPEED 3	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D3S	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D3	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D2Hs	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D2H	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D1H	-	APS-C	CCD	Sony	2.7	5	1005	96%
D810	EXPEED 4	Full-frame	CMOS	Sony	36.3	51	91000	100%
D800	EXPEED 3	Full-frame	CMOS	Sony	36.3	51	91000	100%
D700	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	95%
D750	EXPEED 4	Full-frame	CMOS	Nikon[citation ne...	24.9	51	91000	100%
Df	EXPEED 3	Full-frame	CMOS	Nikon	16.2	39	2016	100%

43 / 43

WIKIDATA

Réponses

- Valider l'implémentation (tester)
- Valider les exigences et l'implémentation à chaque itération
  - Sortie de "release" avec procédure de tests automatisée (git + Jenkins + Junit + PhantomJS)
  - Validation de chaque release avec le client
- Eliciter et modéliser les exigences/besoins avec le client

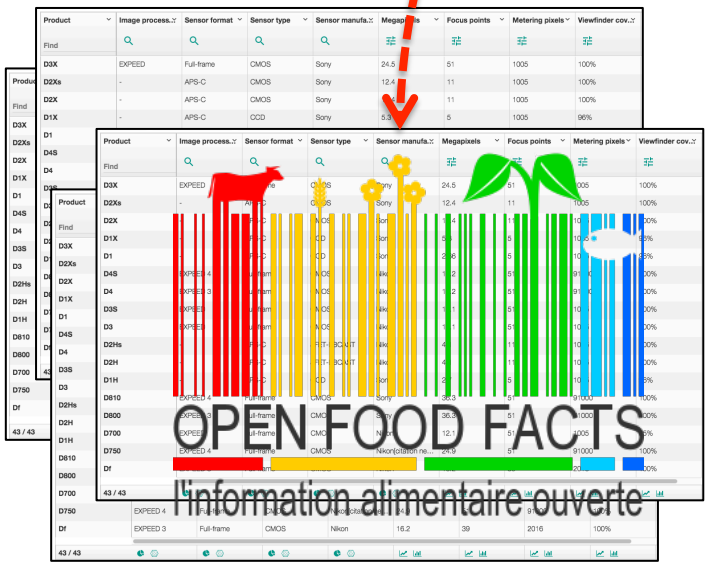
Valider l'implémentation (tests automatisés)

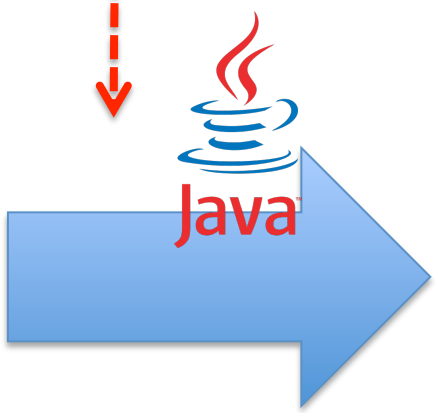
Tests

(sur les entrées)

(sur la transformation)

(sur la sortie)



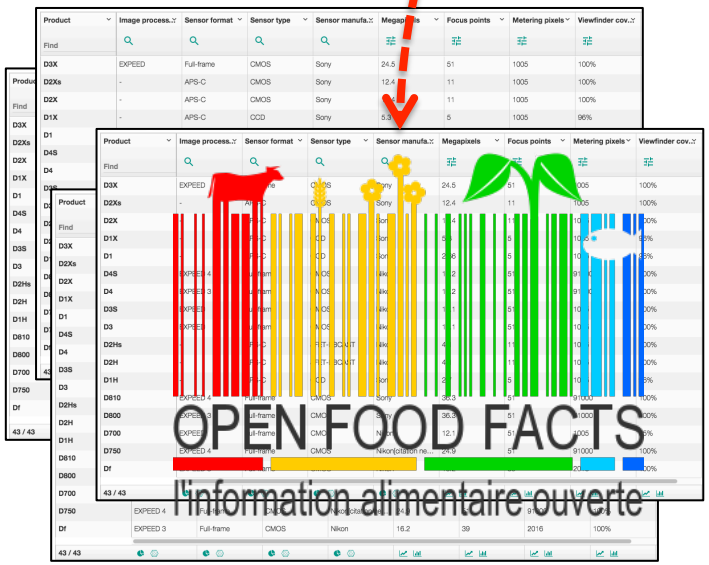


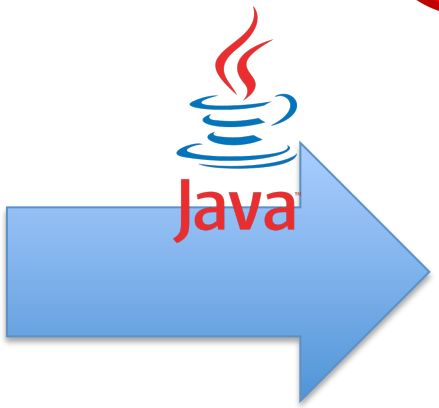
example

CSV

Tests

(sur les entrées)



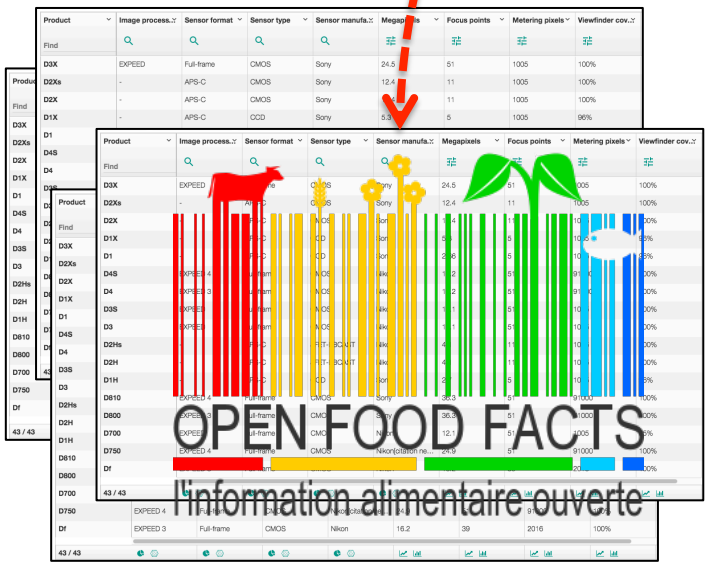


example

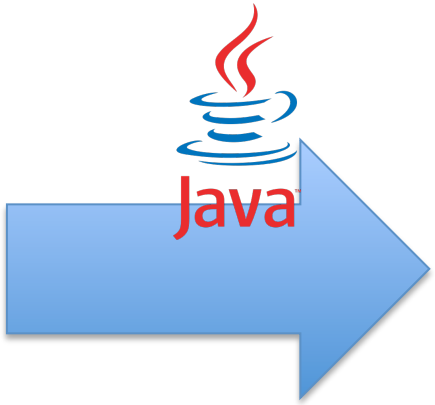
CSV

Tests

(sur les entrées)



RuntimeException….

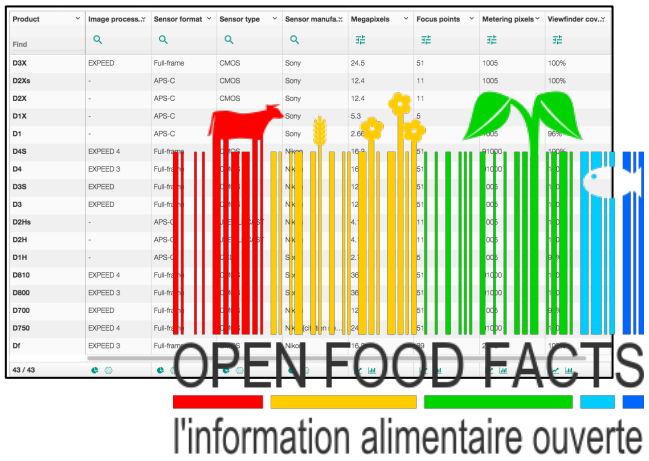


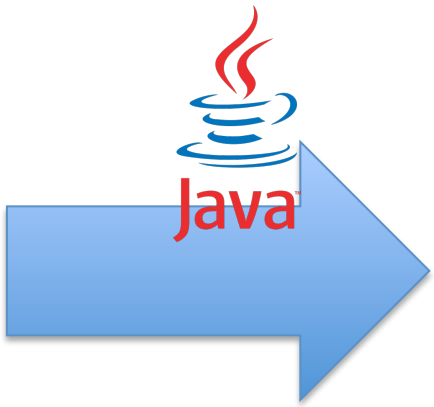
example

CSV

Tests

(sur les entrées)

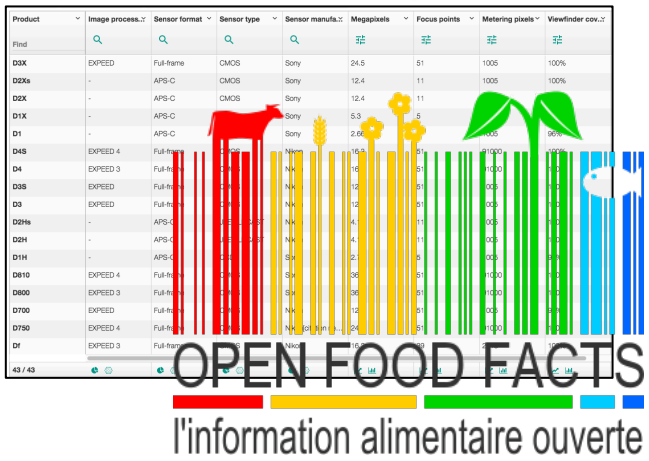


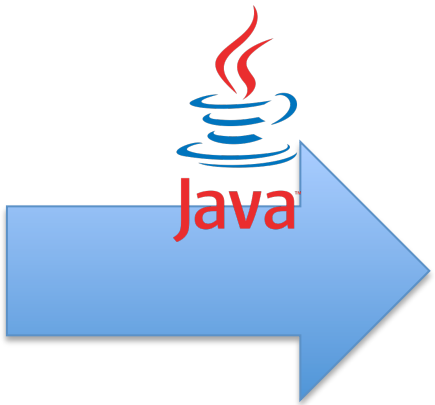


F1;f2; , ;
“”.” ”.
, , ,

Tests

(sur les entrées)





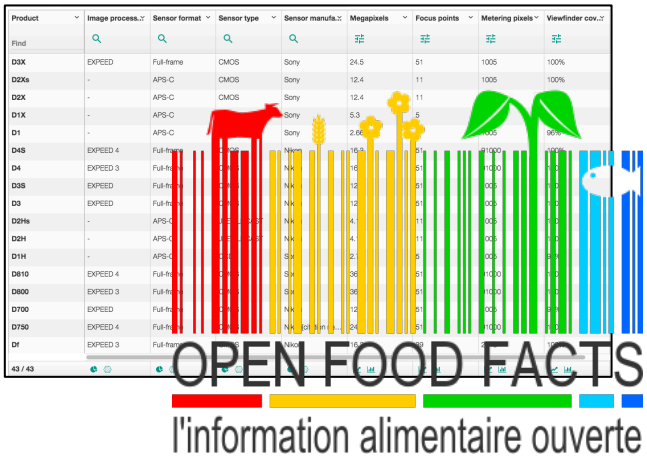
```
f1,f2,f3
v11,v12,v13
v21,v22,v23
```

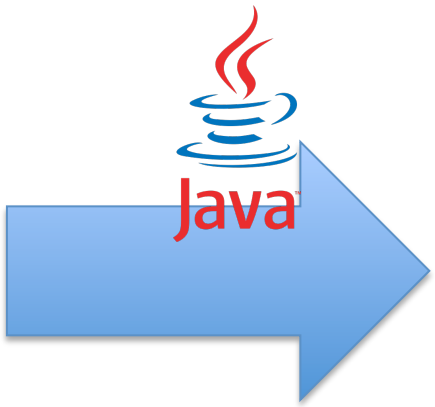
example

CSV

Manual testing is a terrible idea

non reproducible; error-prone; time-consuming



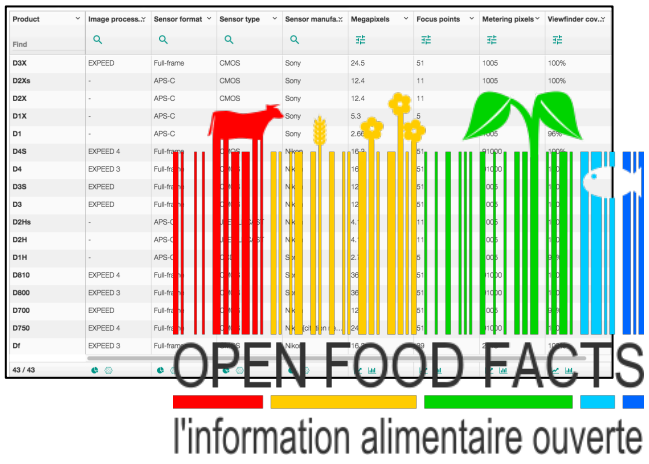


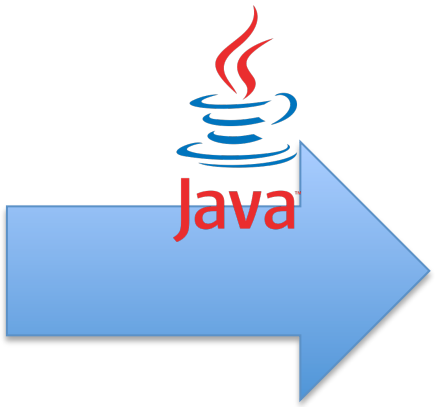
```
f1,f2,f3
v11,v12,v13
v21,v22,v23
```

example

CSV

You can start with some values/ inputs and then (manually) observe





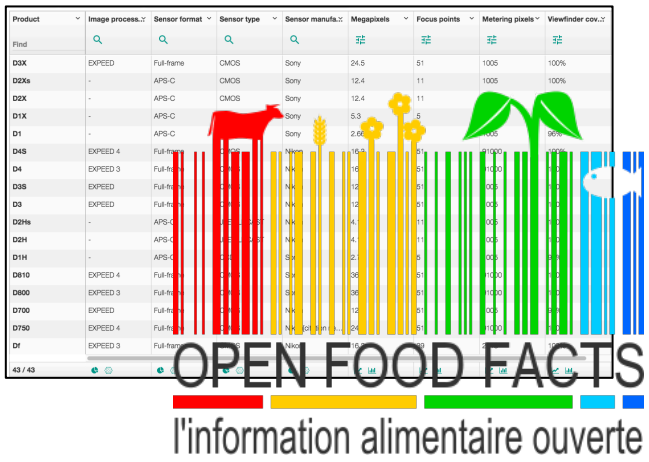
f1,f2,f3
v11,v12,v13
v21,v22,v23

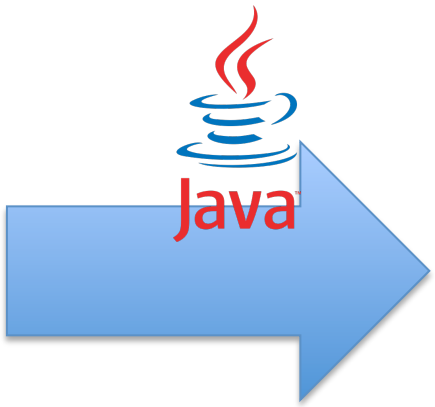
example

CSV

But manual testing is a terrible idea

non reproducible; error-prone; time-consuming





f1,f2,f3
v11,v12,v13
v21,v22,v23

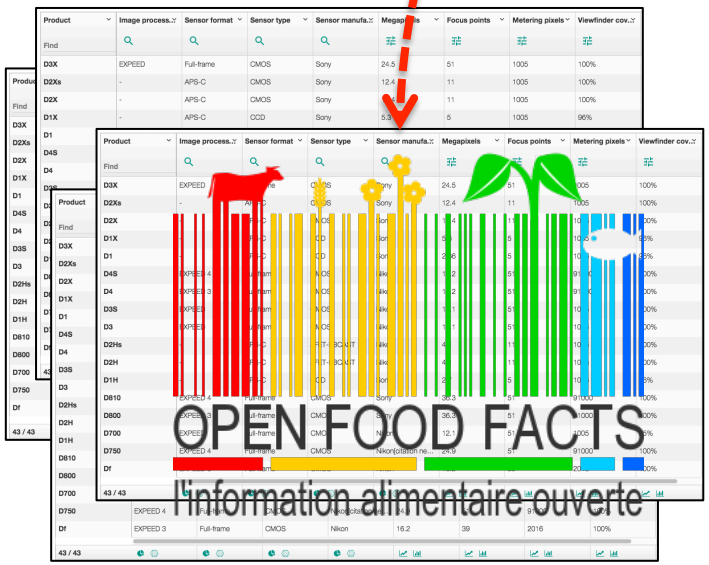
example

CSV

Whenever you are tempted to type something into a print statement or a debugger expression, write it as a test instead.

Tests

(sur les entrées)



Observer par des assertions
(vérification de propriétés)

Java

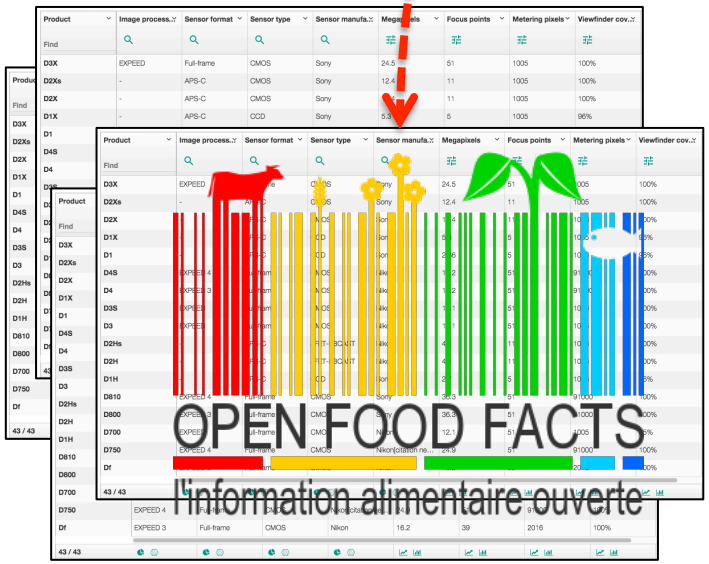
f1,f2,f3
v11,v12,v13
v21,v22,v23

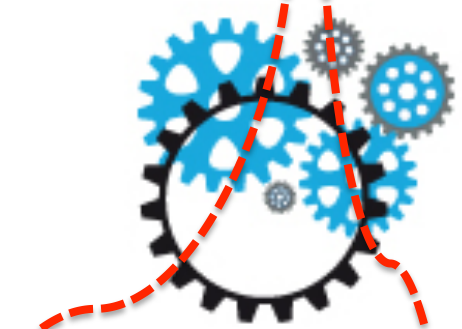
example

CSV

Tests automatisés

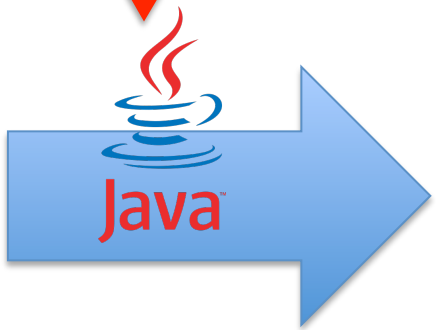
(sur les entrées)





(sur la transformation)

(sur la sortie)

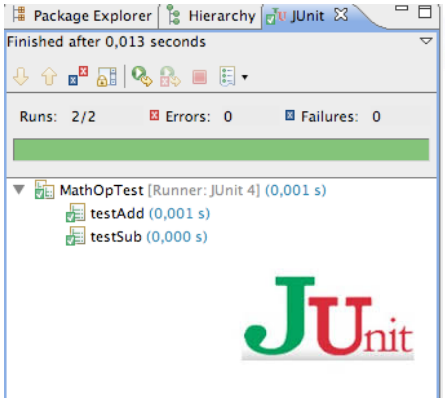


f1,f2,f3
v11,v12,v13
v21,v22,v23

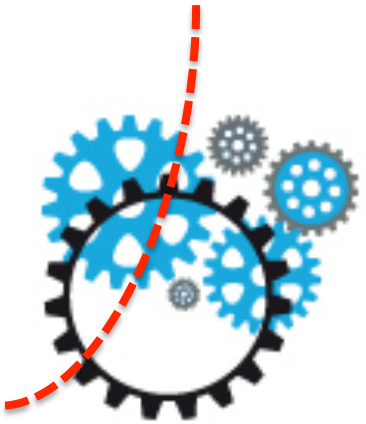
example

CSV

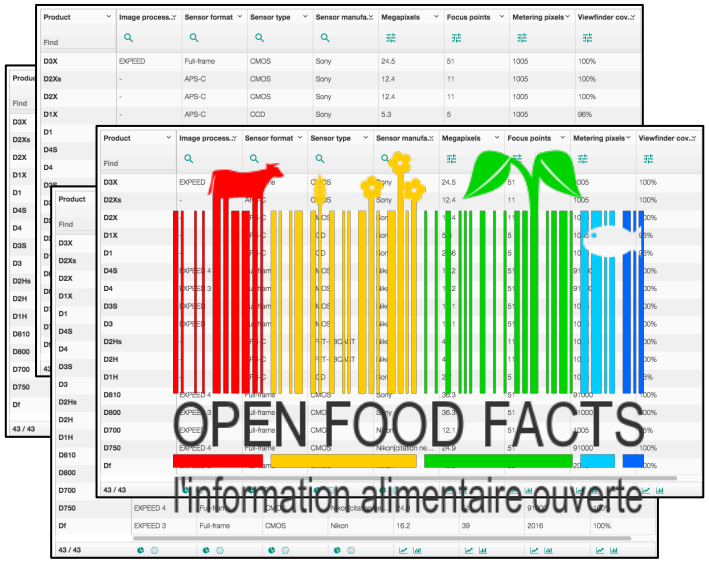
Tests automatisés

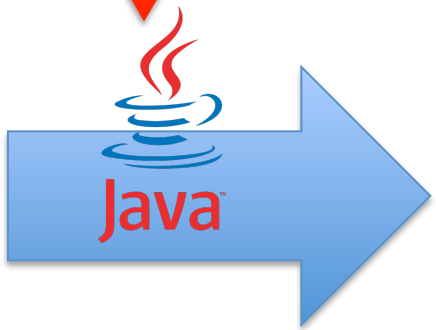



```java
// Tests adding a product to the cart.
public void testProductAdd() {
     Product book = new Product("Refactoring", 53.95);
     _bookCart.addItem(book);

     assertTrue(_bookCart.contains(book));

     double expected = 23.95 + book.getPrice();
     double current = _bookCart.getBalance();

     assertEquals(expected, current, 0.0);

     int expectedCount = 2;
     int currentCount = _bookCart.getItemCount();

     assertEquals(expectedCount, currentCount);
}
```



(sur la transformation)





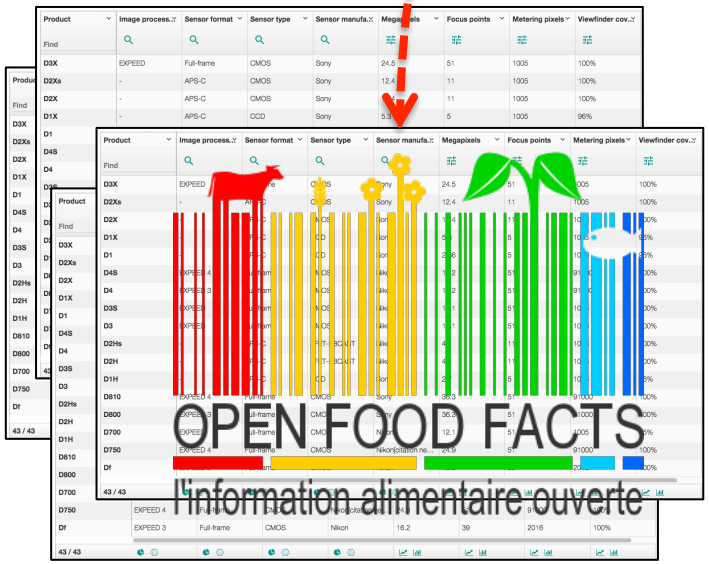
f1,f2,f3
v11,v12,v13
v21,v22,v23

example

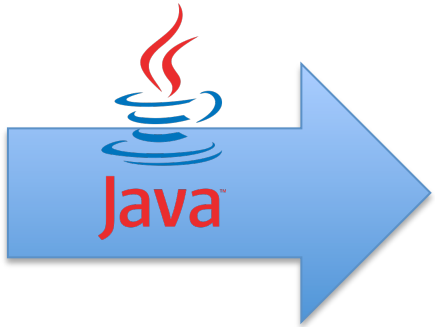
CSV

Tests automatisés

(sur les entrées)



(sur la sortie)



f1,f2,f3
v11,v12,v13
v21,v22,v23

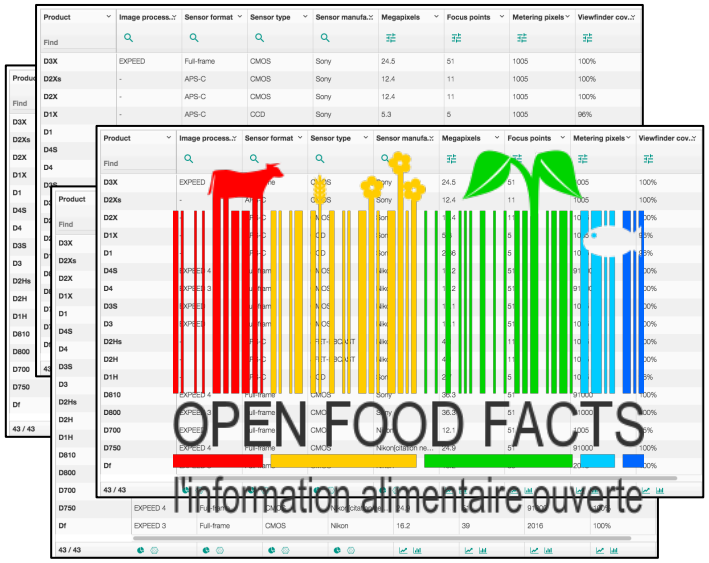
example

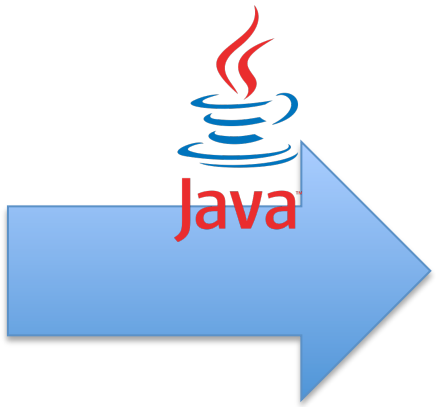
CSV

Tests automatisés

(concevoir un ensemble de données en “input” pertinent pour le test et couvrant un maximum de cas)

(vérifier des assertions « génériques » ou bien produire la sortie attendue puis comparaison aka « diff »)





f1,f2,f3
v11,v12,v13
v21,v22,v23

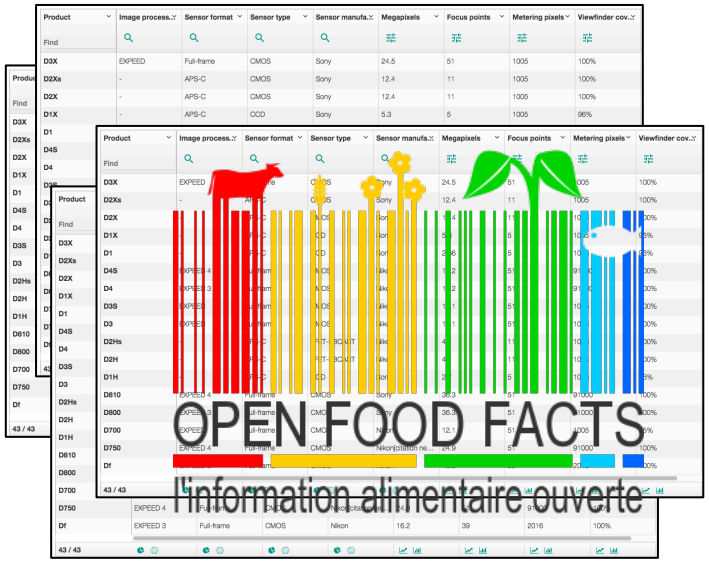
example

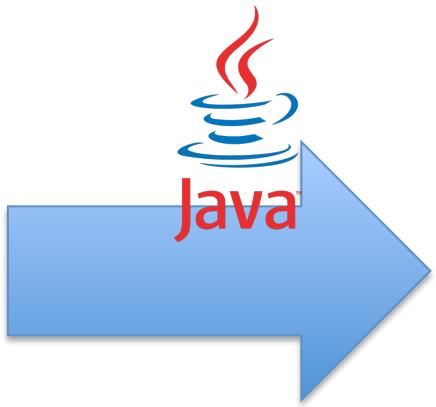
CSV

Tests automatisés

NE PAS TESTER VOTRE SOLUTION SUR UNE SEULE MATRICE!

(vérifier des assertions « génériques » ou bien produire la sortie attendue puis comparaison aka « diff »)



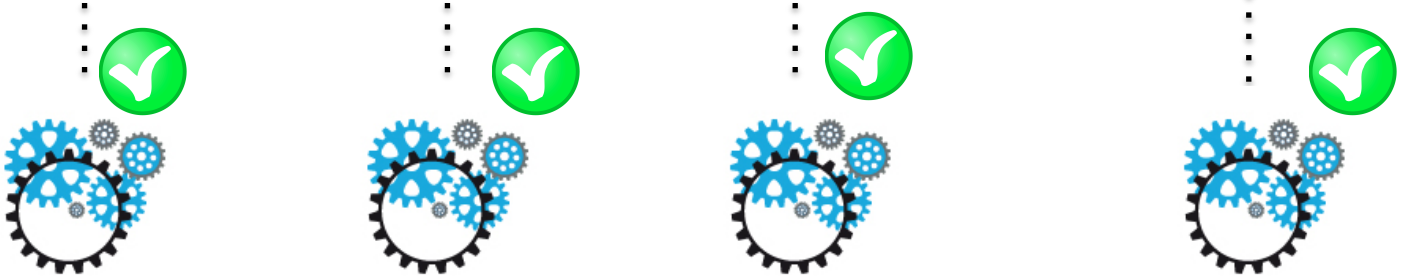


f1,f2,f3
v11,v12,v13
v21,v22,v23

example

CSV

SP (sprints; implémentation)

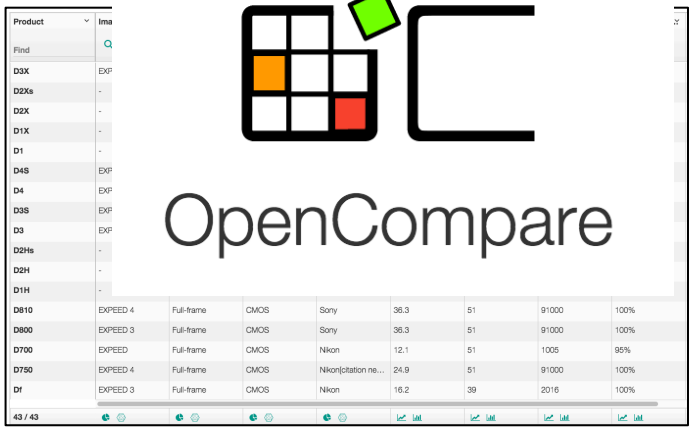


Execute the tests before/after each commit
Don’t break (no regression)
Continuous validation

Tests et projets (bis, discussions)

Projet #1

OpenCompareReverseJSON



OpenCompare

- Java API (incl. JSON parsing)
- JSON Schema
- UML Class diagram

Projet #2

OpenFoodFacts2CSV

OPEN FOOD FACTS
l'information alimentaire ouverte

CSV
(Comma Separated Values)

Product	Image process.	Sensor format	Sensor type	Sensor manufa.	Megapixels	Focus points	Metering pixels	Viewfinder cov.
Find								
D3X	EXPEED	Full-frame	CMOS	Sony	24.5	51	1005	100%
D2Xs	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D2X	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D1X	-	APS-C	CCD	Sony	5.3	5	1005	96%
D1	-	APS-C	CCD	Sony	2.66	5	1005	96%
D4S	EXPEED 4	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D4	EXPEED 3	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D3S	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D3	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D2Hs	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D2H	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D1H	-	APS-C	CCD	Sony	2.7	5	1005	96%
D810	EXPEED 4	Full-frame	CMOS	Sony	36.3	51	91000	100%
D800	EXPEED 3	Full-frame	CMOS	Sony	36.3	51	91000	100%
D700	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	96%
D750	EXPEED 4	Full-frame	CMOS	Nikon[citation ne...	24.9	51	91000	100%
Df	EXPEED 3	Full-frame	CMOS	Nikon	16.2	39	2016	100%
43 / 43								

OpenCompare

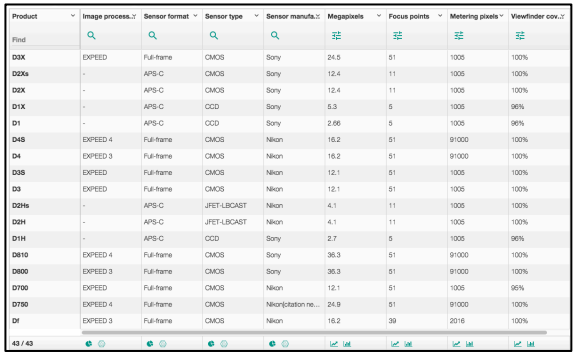
Projet #3

MatrixSynthesizerWikipedia

WIKIDATA

WIKIPEDIA
The Free Encyclopedia

CSV
(Comma Separated Values)



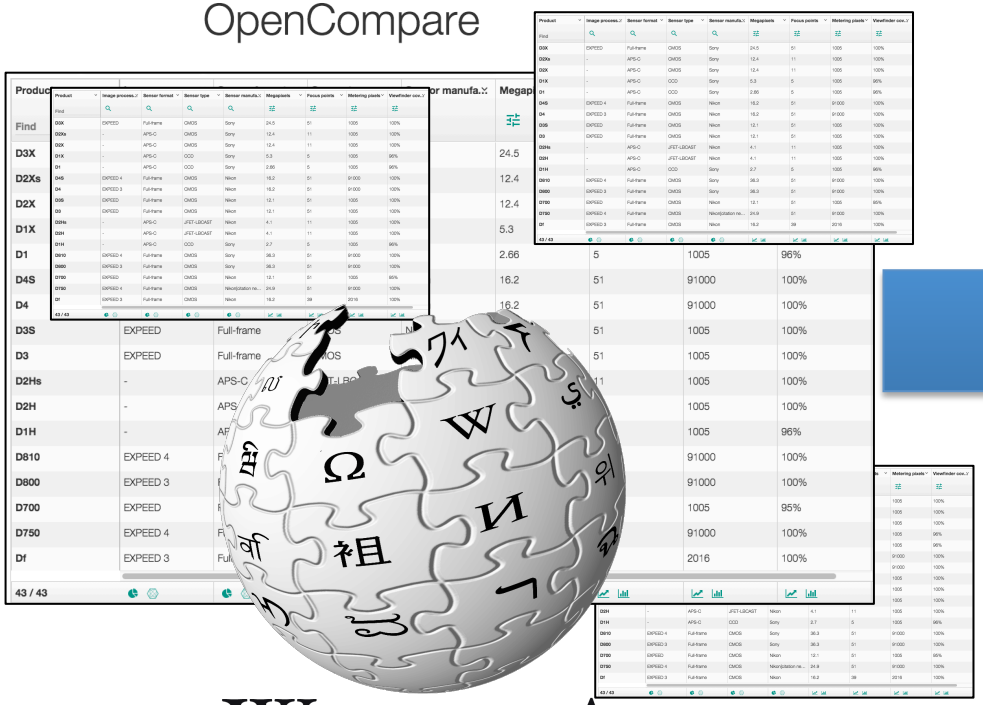
Product	Image process..	Sensor format	Sensor type	Sensor manufa..	Megapixels	Focus points	Metering pixels	Viewfinder cov..
Find								
D3X	EXPEED	Full-frame	CMOS	Sony	24.5	51	1005	100%
D2Xs	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D2X	-	APS-C	CMOS	Sony	12.4	11	1005	100%
D1X	-	APS-C	CCD	Sony	5.3	5	1005	96%
D1	-	APS-C	CCD	Sony	2.66	5	1005	96%
D4S	EXPEED 4	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D4	EXPEED 3	Full-frame	CMOS	Nikon	16.2	51	91000	100%
D3S	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D3	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	100%
D2Hs	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D2H	-	APS-C	JFET-LBCAST	Nikon	4.1	11	1005	100%
D1H	-	APS-C	CCD	Sony	2.7	5	1005	96%
D810	EXPEED 4	Full-frame	CMOS	Sony	36.3	51	91000	100%
D800	EXPEED 3	Full-frame	CMOS	Sony	36.3	51	91000	100%
D700	EXPEED	Full-frame	CMOS	Nikon	12.1	51	1005	95%
D750	EXPEED 4	Full-frame	CMOS	Nikon[citation ne...	24.9	51	91000	100%
Df	EXPEED 3	Full-frame	CMOS	Nikon	16.2	39	2016	100%
43 / 43								

OpenCompare

Projet #4

WikipediaMatrixAnalysis

OpenCompare

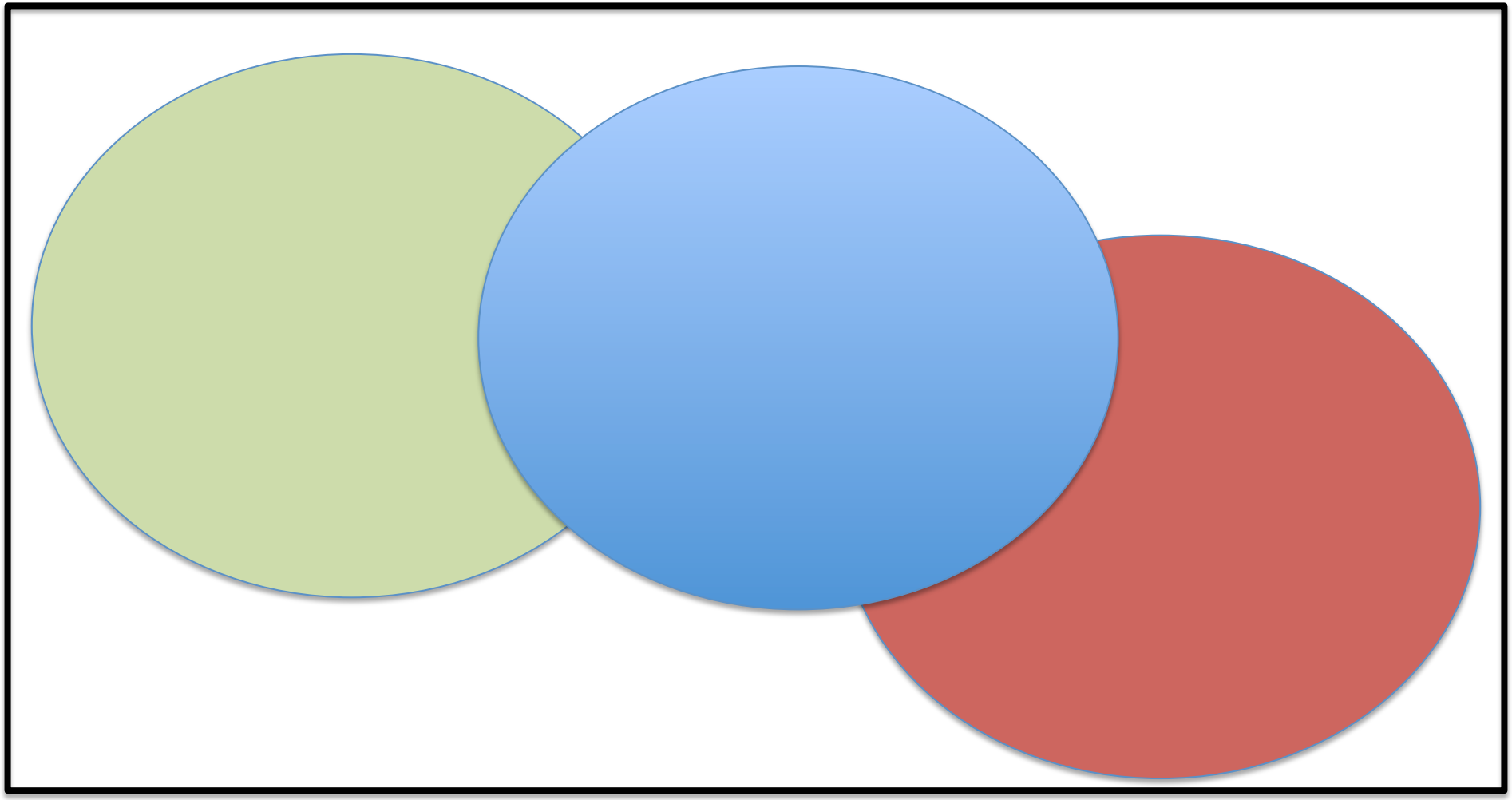


WIKIPEDIA
The Free Encyclopedia

Automated
Analysis

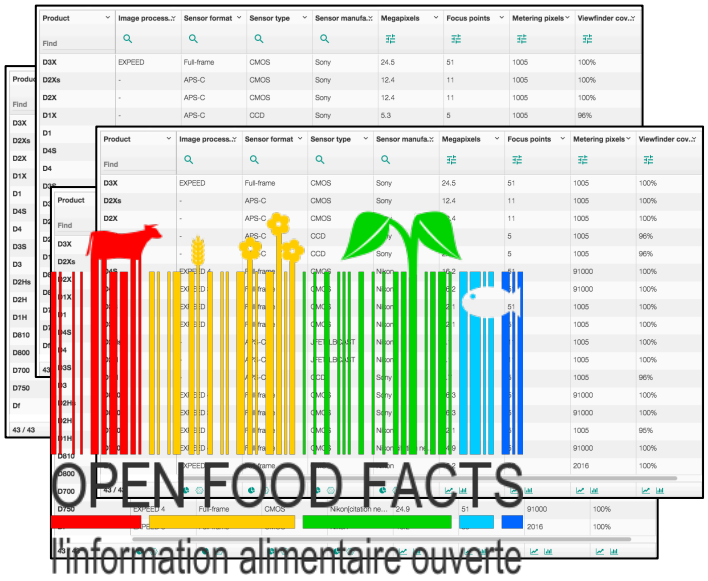
Modéliser les exigences

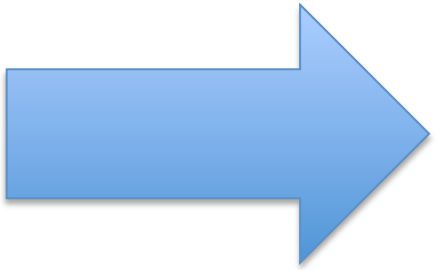
Implémenter des fonctionnalités inutiles ou qui ne correspondent pas aux besoins du client => **Modéliser les exigences**



(à côté de la plaque?)

Modéliser les exigences





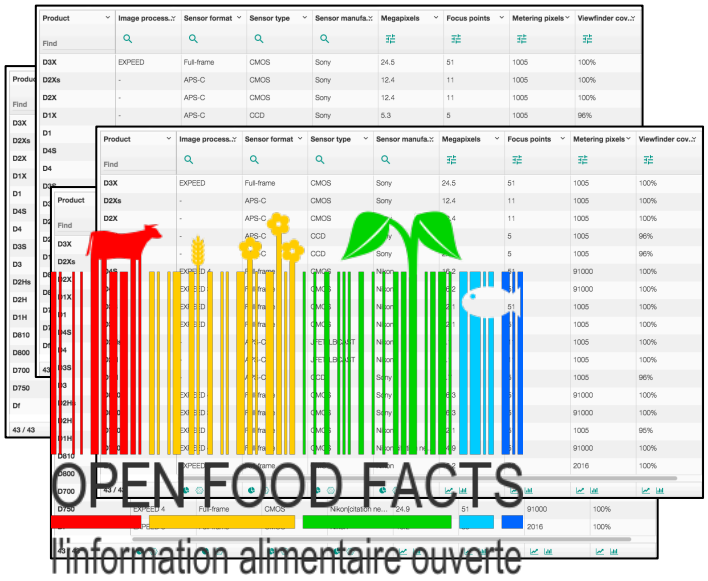
f1,f2,f3
v11,v12,v13
v21,v22,v23

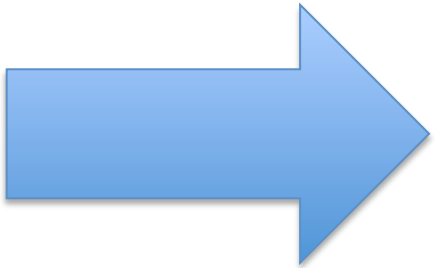
example

CSV

Expliciter
Documenter
Communiquer avec le client

Modéliser les exigences





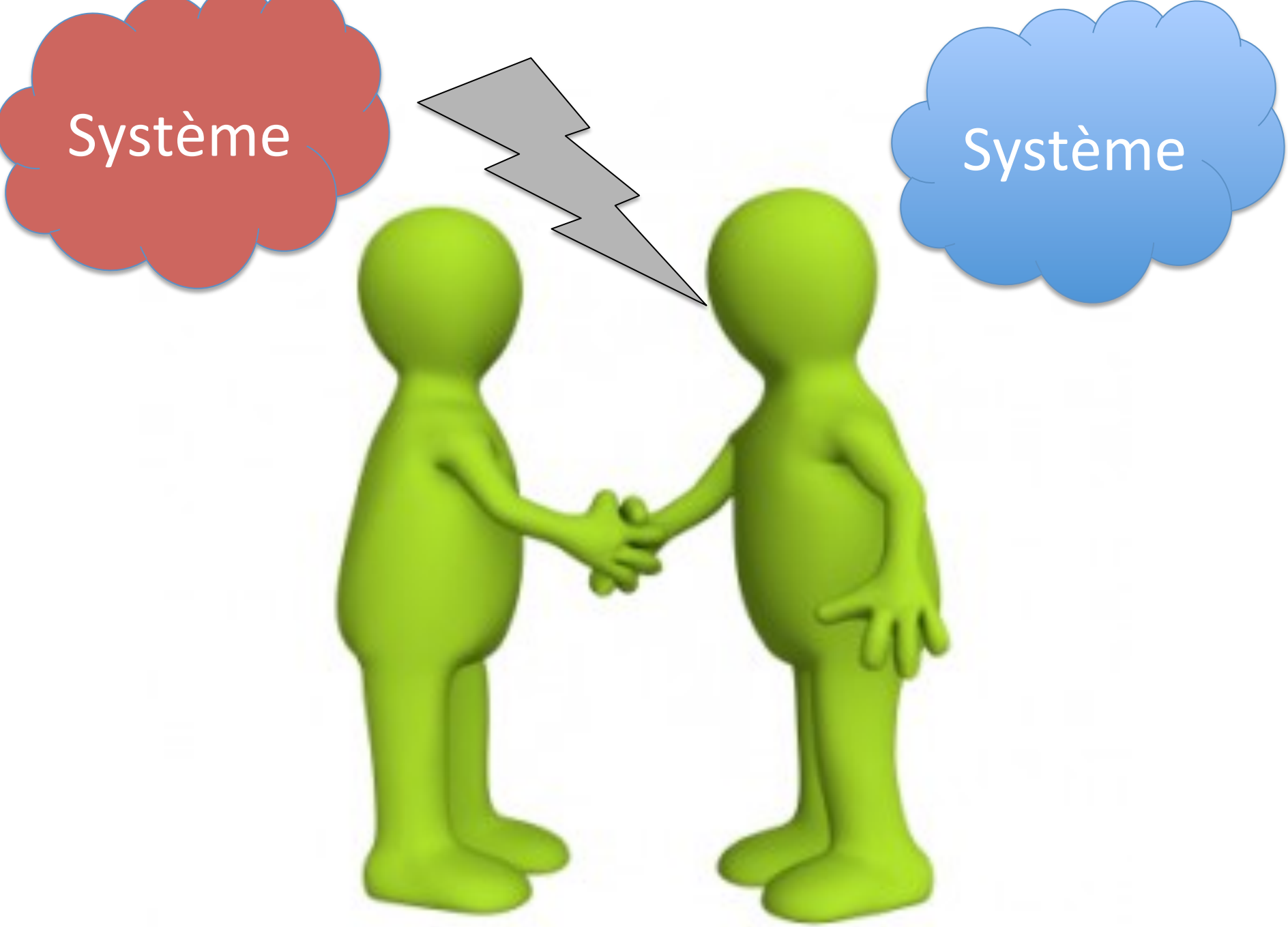
f1,f2,f3
v11,v12,v13
v21,v22,v23

example

CSV

Délimiter
Approche défensive
“Contrat”

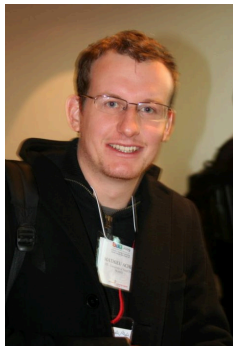
Valider les exigences et
l'implémentation
à chaque itération
avec le client

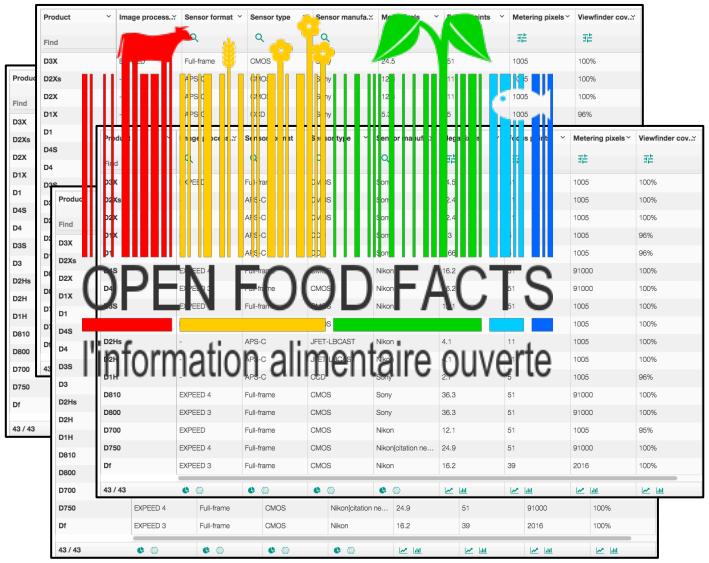

Système
Système

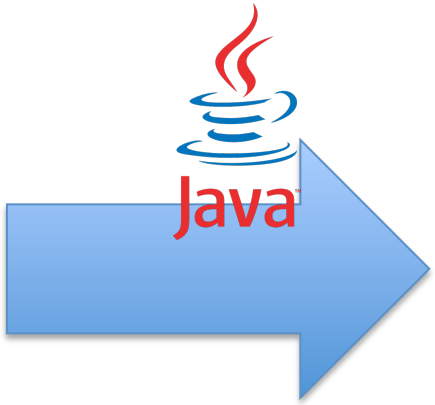
Validation par le client et/ou des utilisateurs

NE PAS TESTER VOTRE SOLUTION SUR UNE SEULE MATRICE!

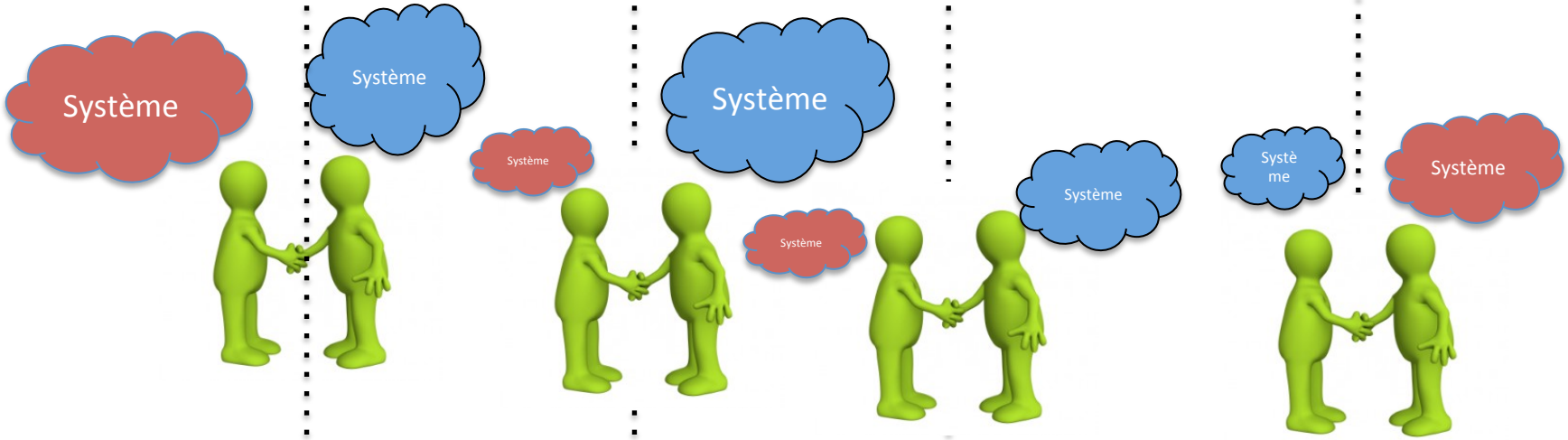
"la stratégie de filtrage sur OFF n'est pas correcte"
"la librairie XYZ n'est pas adaptée"



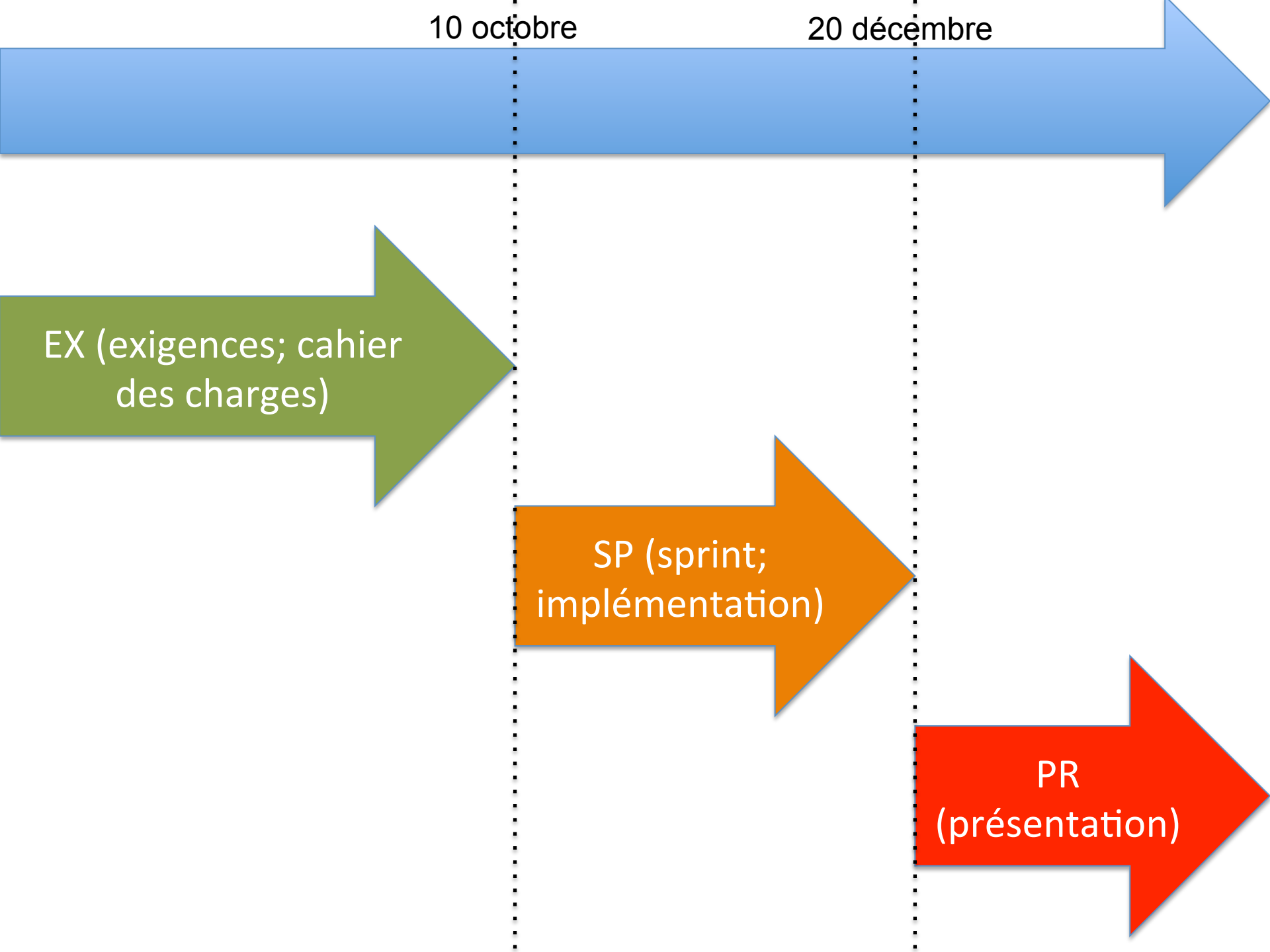


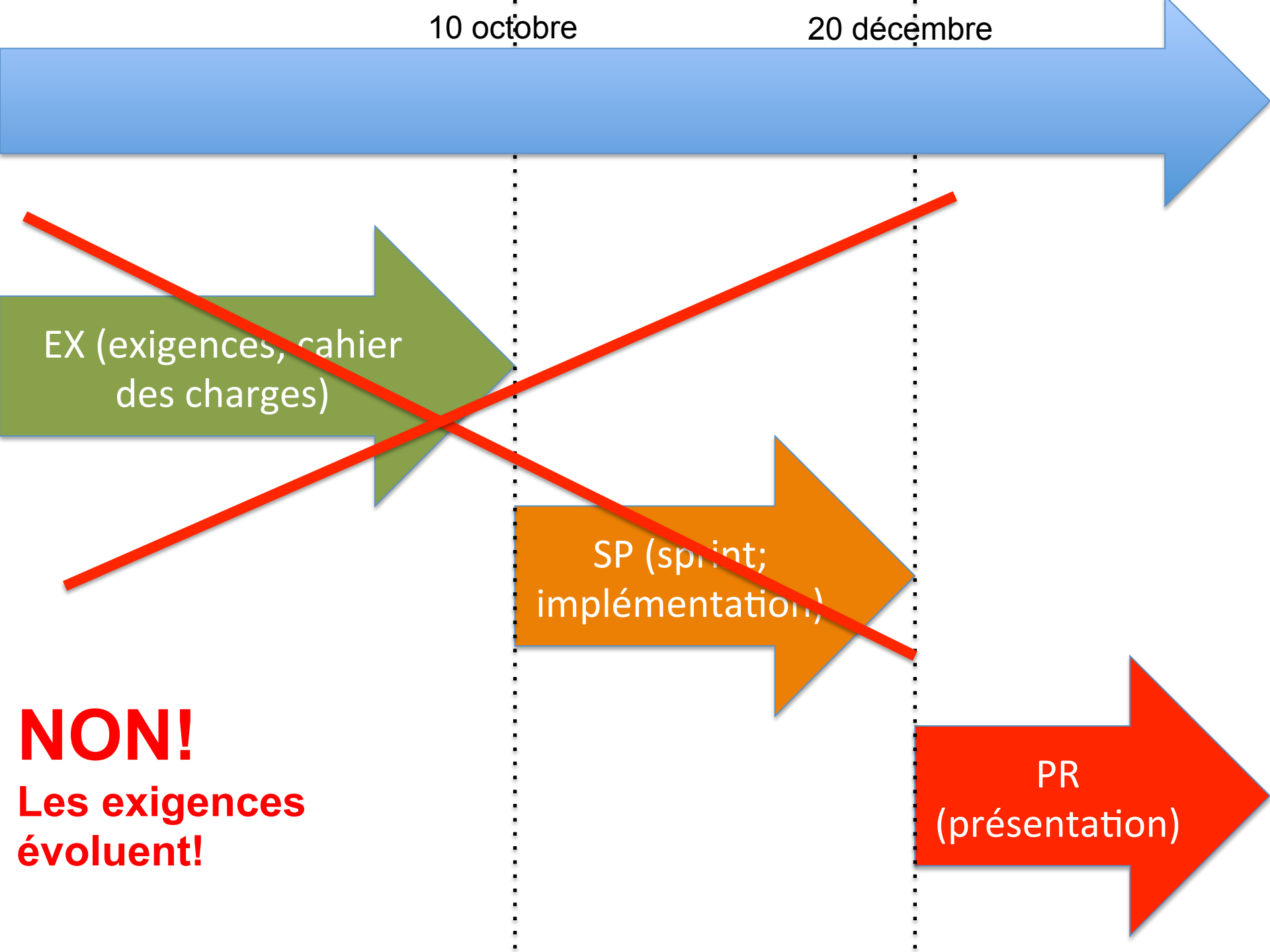


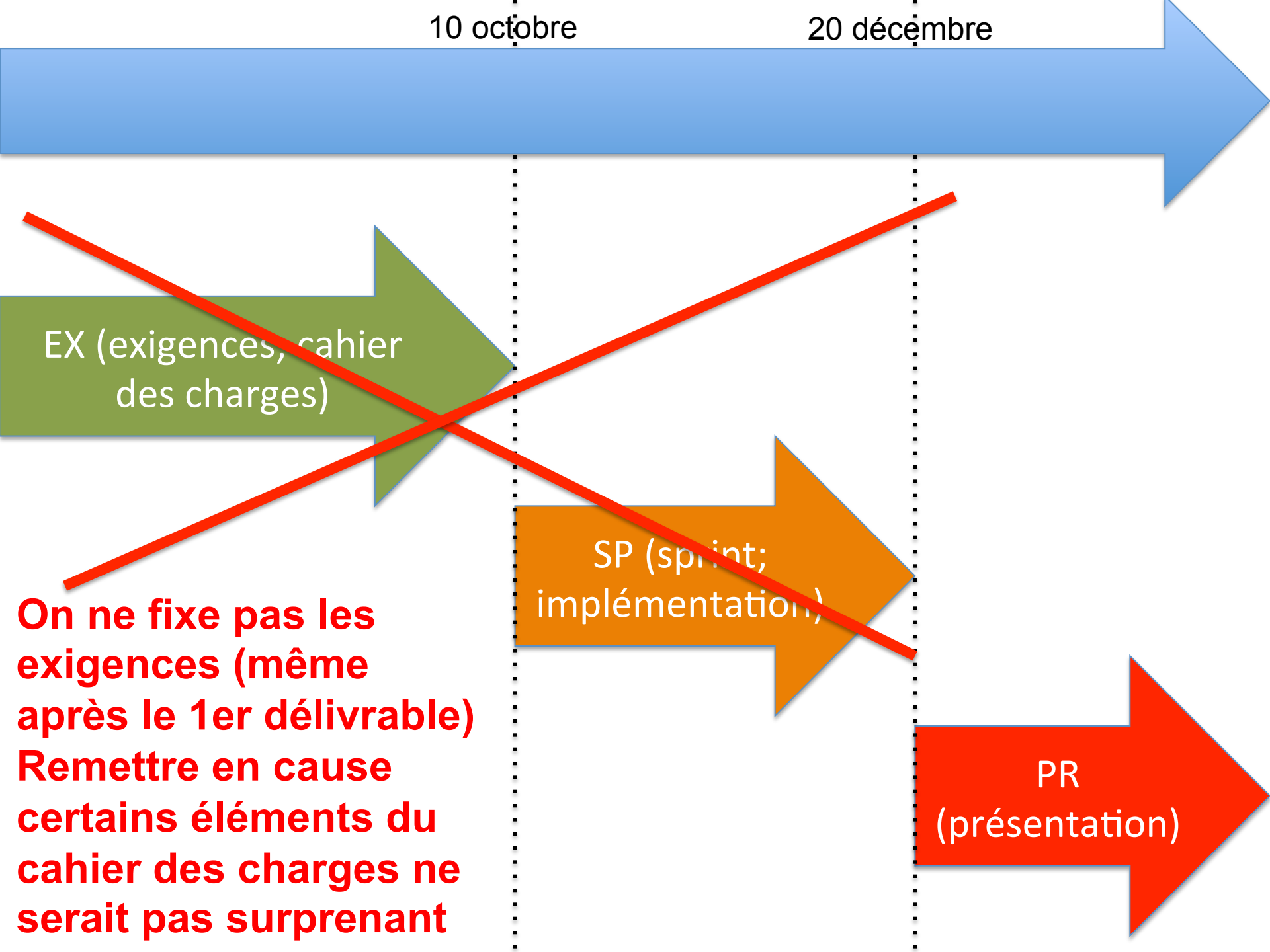
EX (exigences; cahier des charges)

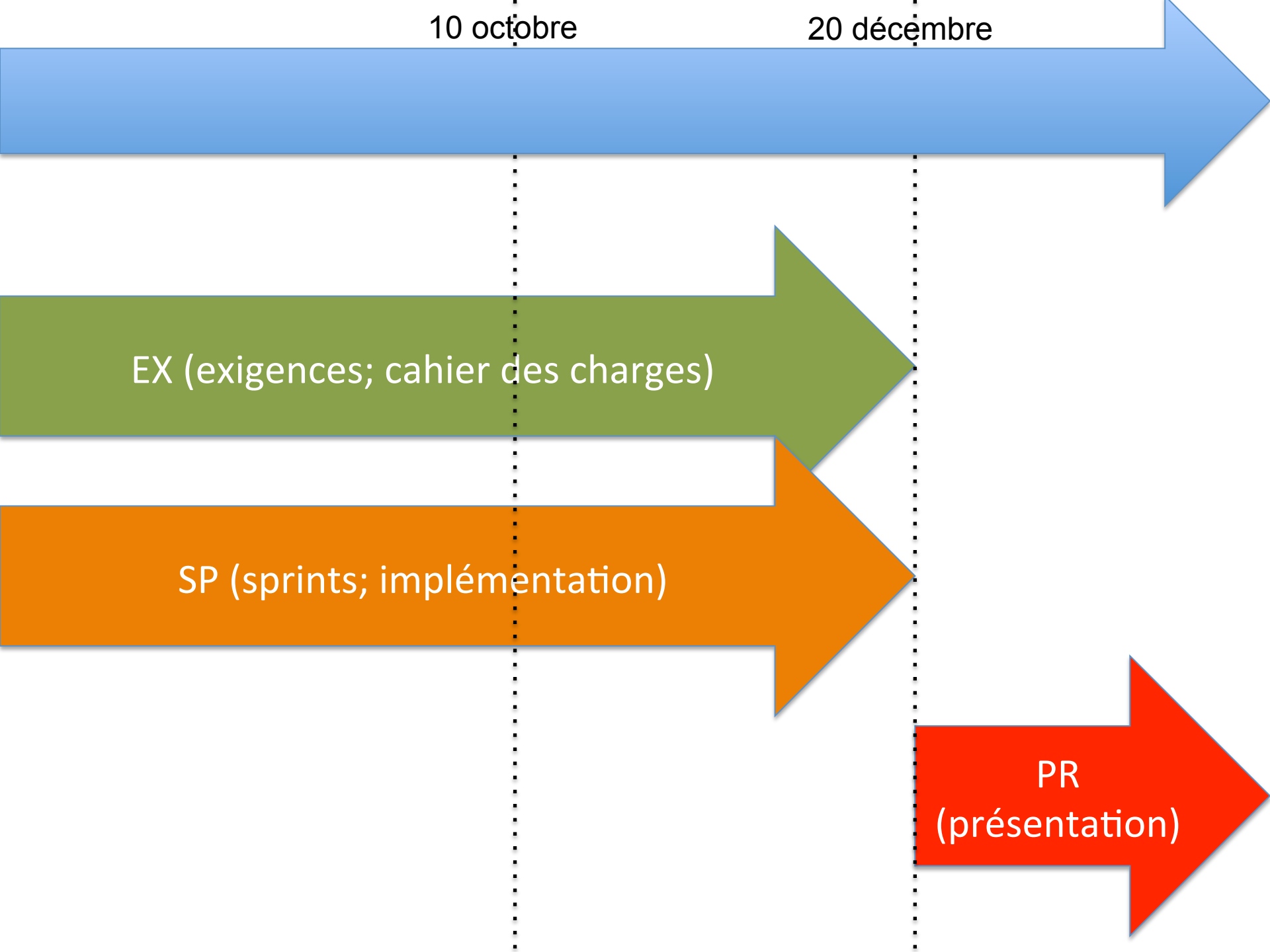


Valider à chaque itération avec le client: montrer les modèles, expliquer les choix technologiques, etc.


10 octobre
20 décembre
EX (exigences; cahier des charges)
SP (sprint; implémentation)
PR (présentation)

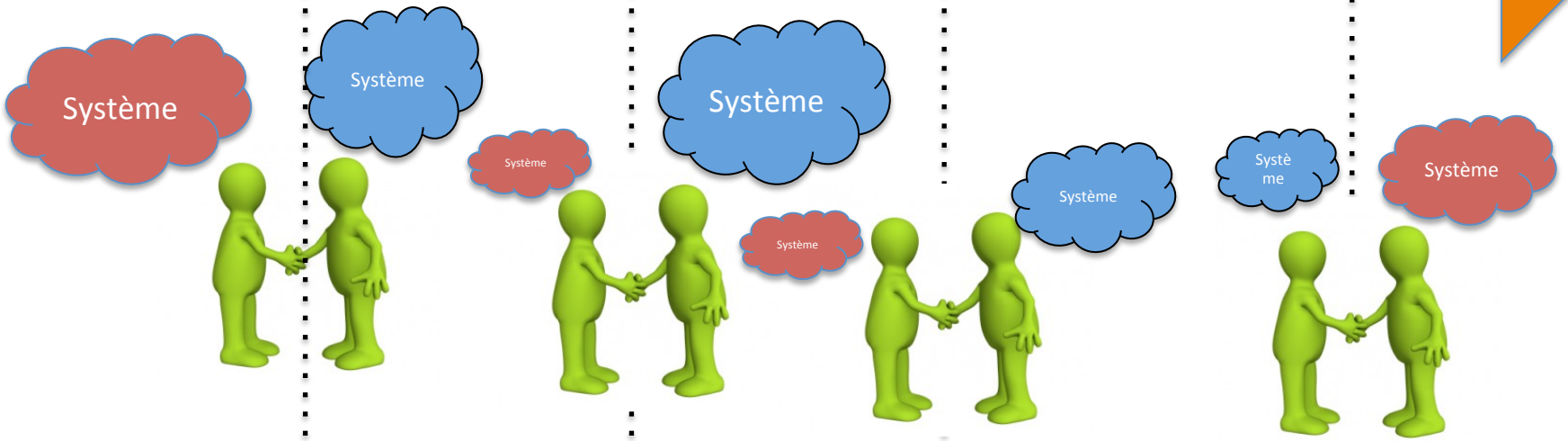

10 octobre
20 décembre
EX (exigences, cahier des charges)
SP (sprint; implémentation)
PR (présentation)
NON!
Les exigences évoluent!


10 octobre
20 décembre
EX (exigences, cahier des charges)
SP (sprint; implémentation)
PR (présentation)
On ne fixe pas les exigences (même après le 1er délivrable)
Remettre en cause certains éléments du cahier des charges ne serait pas surprenant

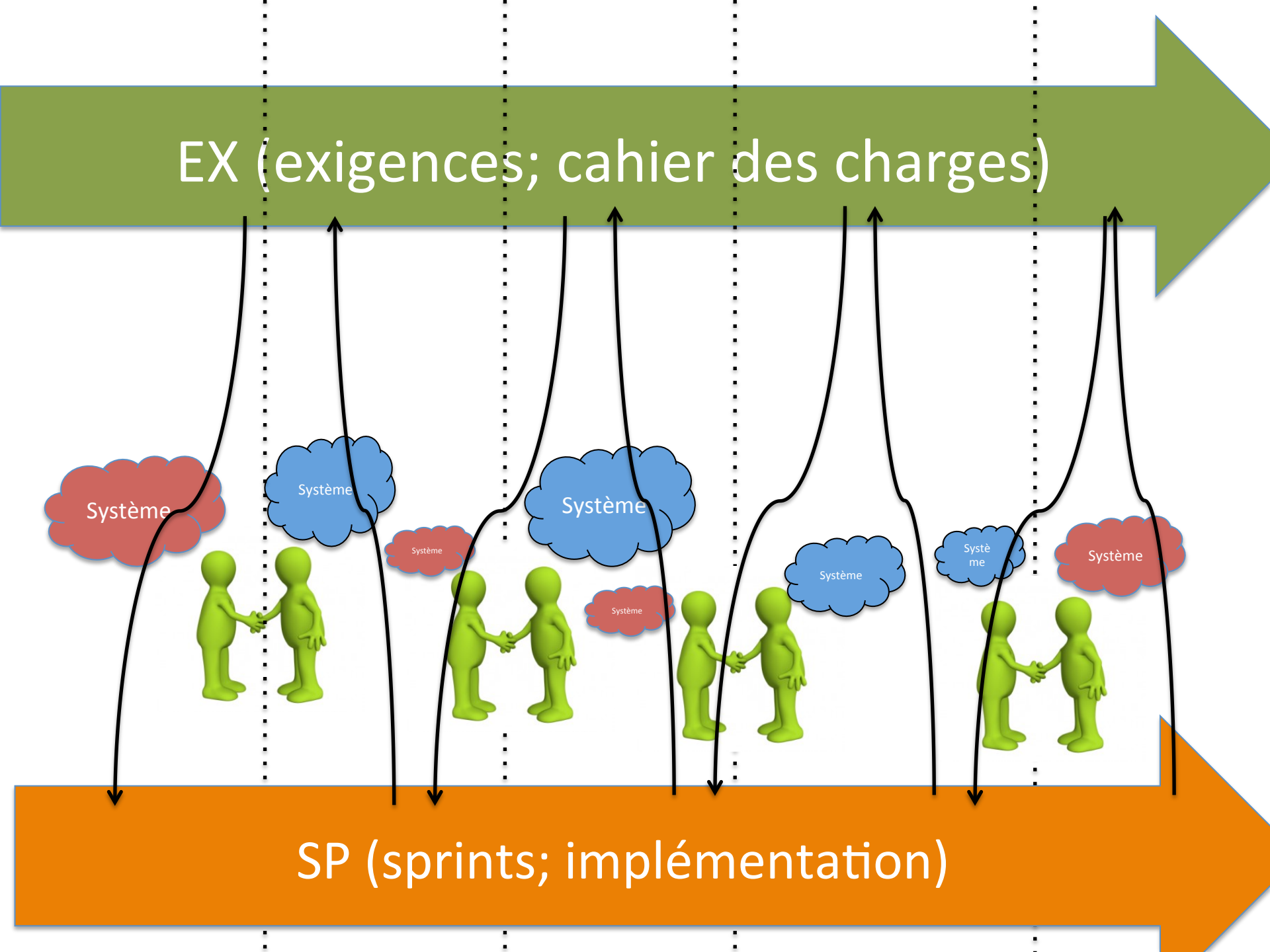

10 octobre
20 décembre
EX (exigences; cahier des charges)
SP (sprints; implémentation)
PR
(présentation)

EX (exigences; cahier des charges)

SP (sprints; implémentation)



Valider à chaque itération avec le client: montrer les exigences et l'implémentation (le « produit » en action)


EX (exigences; cahier des charges)
Système
Système
Système
Système
Système
Système
Système
Système
SP (sprints; implémentation)

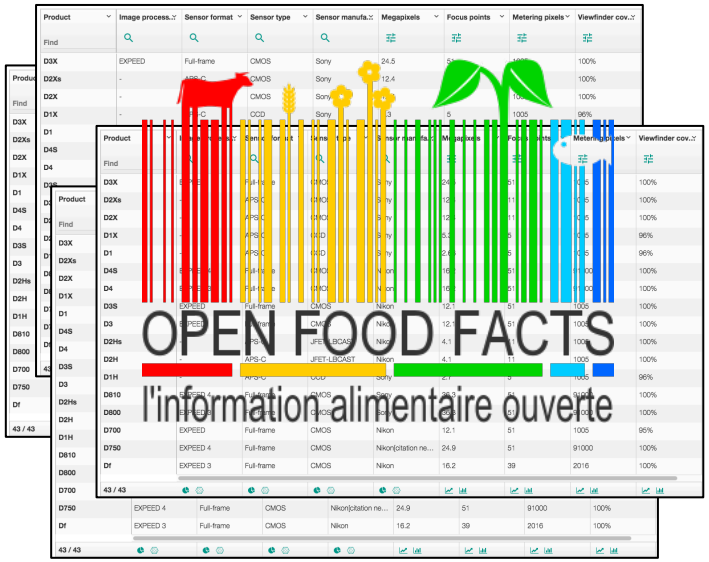
Conclusion

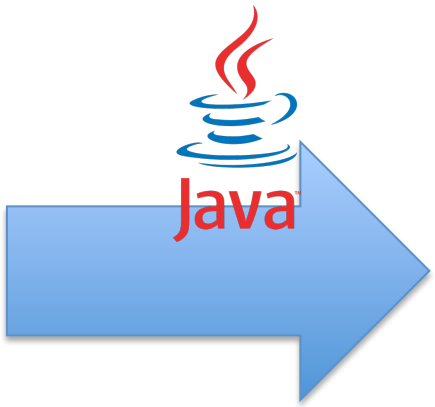
En résumé

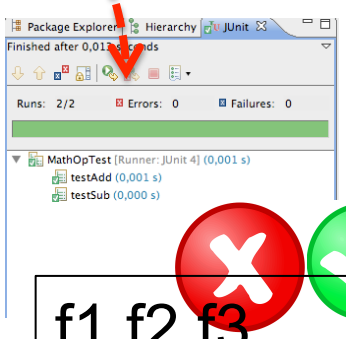
- Modéliser les exigences/besoins avec le client de manière continue (EX)
- Valider l'implémentation par le test (SP)
- Valider les exigences et l'implémentation à chaque itération avec le client ou des utilisateurs
  - Montrer le "produit" en action permet de raffiner les exigences
- Conséquence: sorties fréquentes de "release"
  - Solution: procédure de tests automatisée (git + CI + Junit)

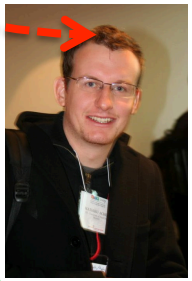
#1 tests automatiques (exhaustif)
#2 validation par le client/des utilisateurs (sampling)

NE PAS TESTER VOTRE SOLUTION SUR UNE SEULE MATRICE!







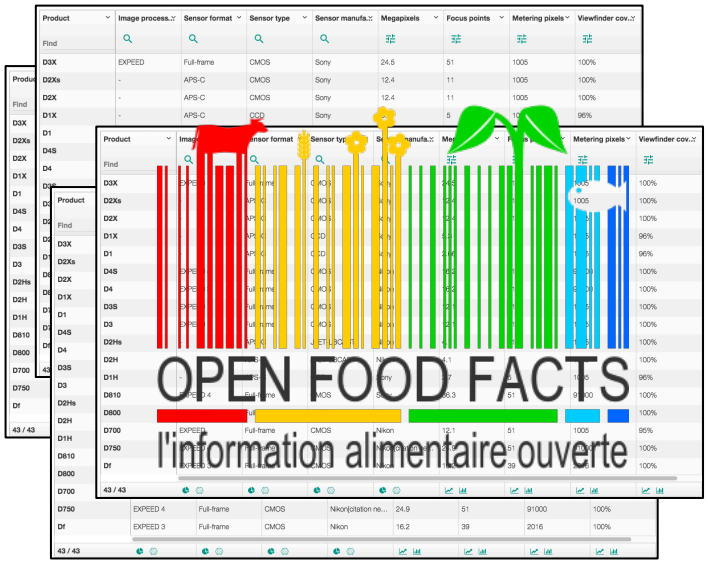


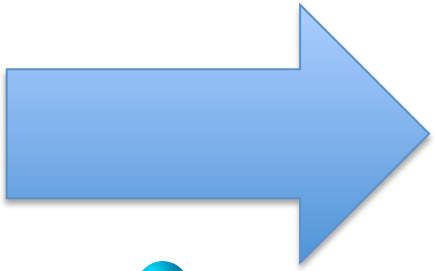
f1,f2,f3
v11,v12,v13
v21,v22,v23

example

CSV

Travail collaboratif et itératif (multi-persons, multi-versions)





f1,f2,f3
v11,v12,v13
v21,v22,v23

example

CSV

Multi-Tools and Languages