

One Year as a (transition) CIO



Pierre Pezziardi
Lean IT Summit
13th october 2011

EYROLLES
Éditions d'Organisation



USI 2011 - 28 & 29 juin - Paris - Casablanca
Pour une informatique qui transforme nos sociétés

L'événement USI

Philosophie
de l'événement

USI
en 15 minutes

Création
du club USI

«Que faisons-nous aujourd'hui quand nous spécifions un système ? Nous observons ce que fait l'utilisateur, comment il opère, et ensuite nous écrivons un système qui l'aide. Nous prenons les règles, nous les injectons dans le logiciel, et désormais ces anciennes règles sont gravées dans le marbre ! Maintenant il peut faire exactement les mêmes erreurs, mais plus vite !»



Eliyahu Goldratt

Auteur, scientifique, philosophe, père de la



[Facebook](#) [Twitter](#) [Forum](#) [Wiki](#) [Blog](#)

[English](#)



Octopus Microfinance Software

Open-source community for microfinance and banks

babyloan.org
micro credits, great stories

[Home](#) [Product](#) [Services](#) [Download](#) [Support](#) [Contribute](#) [About us](#) [Contact us](#)



WTF?

Entrepreneur, author, lean,
micro-finance & IT



Octopus community news

find all the latest Octopus news on our community blog

BRED

Banque Populaire
3500+ people
Corporate & Retail

Mutualist



IT

230 people

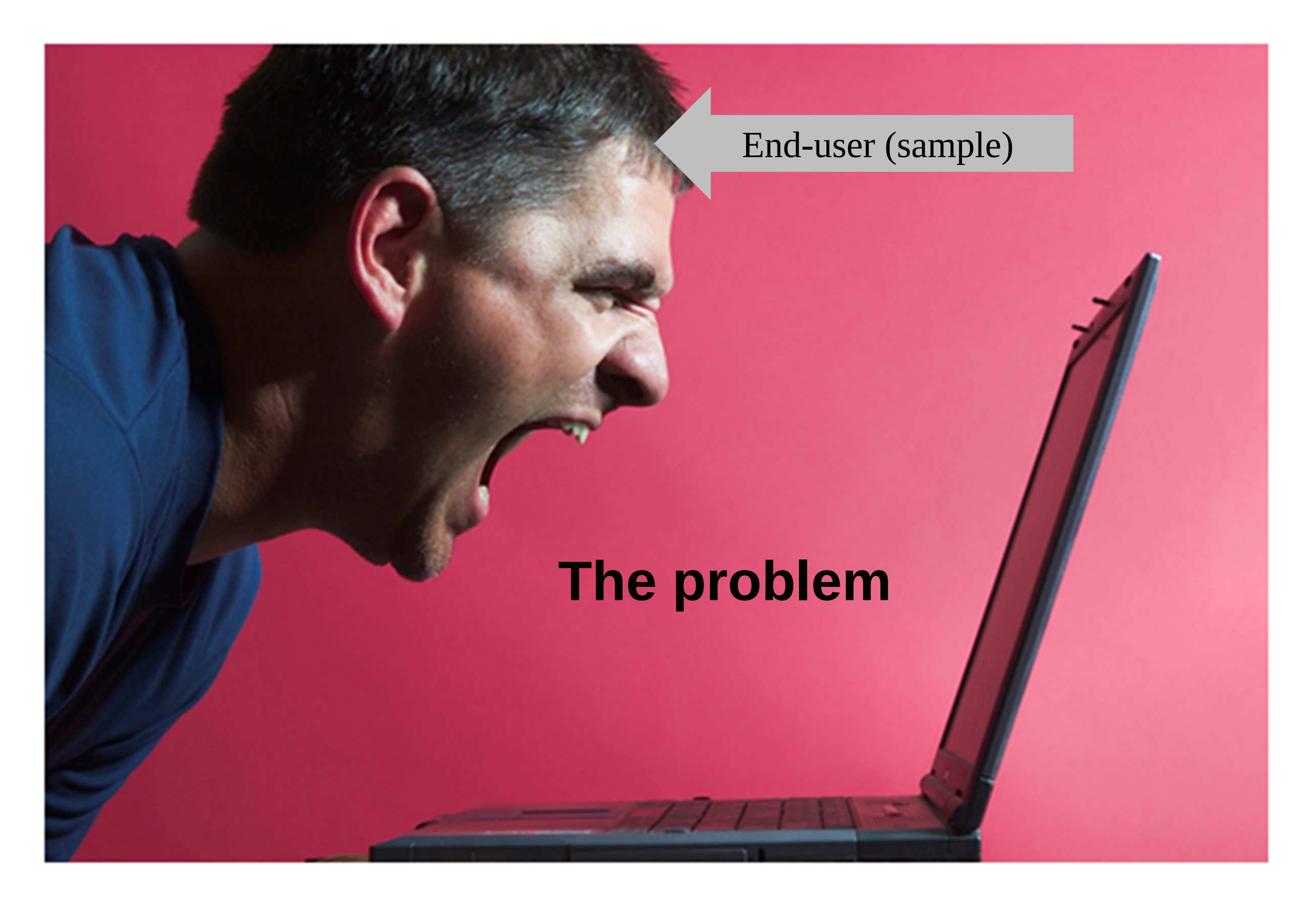
Créteil (Paris suburbs)



Development
(120 p.)

Operations
(70 p.)

Customer
Service
(40 p.)

A photograph of a man in profile, wearing a blue shirt, shouting with his mouth wide open at a laptop screen. The background is a solid red color. A grey arrow points from the text 'End-user (sample)' to the man's head.

End-user (sample)

The problem

5 field experiments

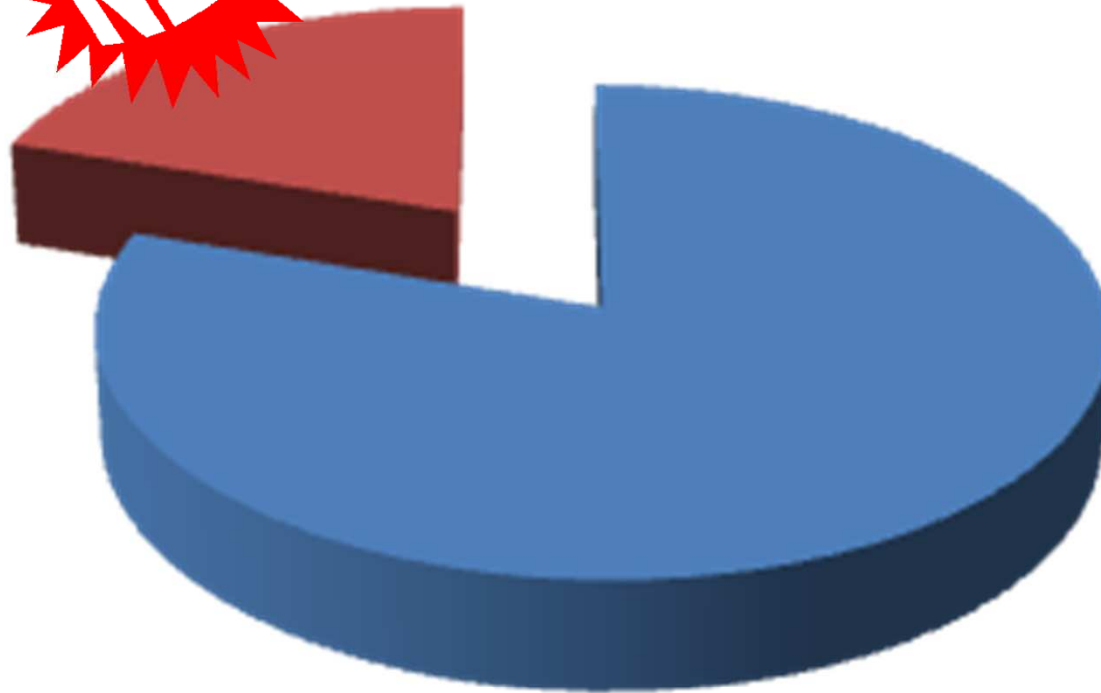
- Let's (try to) **free up energies**
- Bank branch “**bubble**”
- 2M€ fixed price **Agile** project
- **PDCA** at the Customer Service
- **devOps** reorganization

- No silver bullet, small steps repeated
- People can (must) do the change
- Talking about problems is good, suffering in silence is bad



recruit new leaders

NEW



■ Work on things you've been asked for

■ Work on things you personally suffer from, aka waste

A photograph of a rural landscape featuring a grassy hillside. Three cows are grazing: a white cow on the left, a brown cow further back, and a black and white cow in the center. A wire fence with wooden posts runs across the foreground. The background is filled with lush green trees and foliage.

Free up energies ?



Engage only when there is a problem
AND
someone who (really) wants to solve it



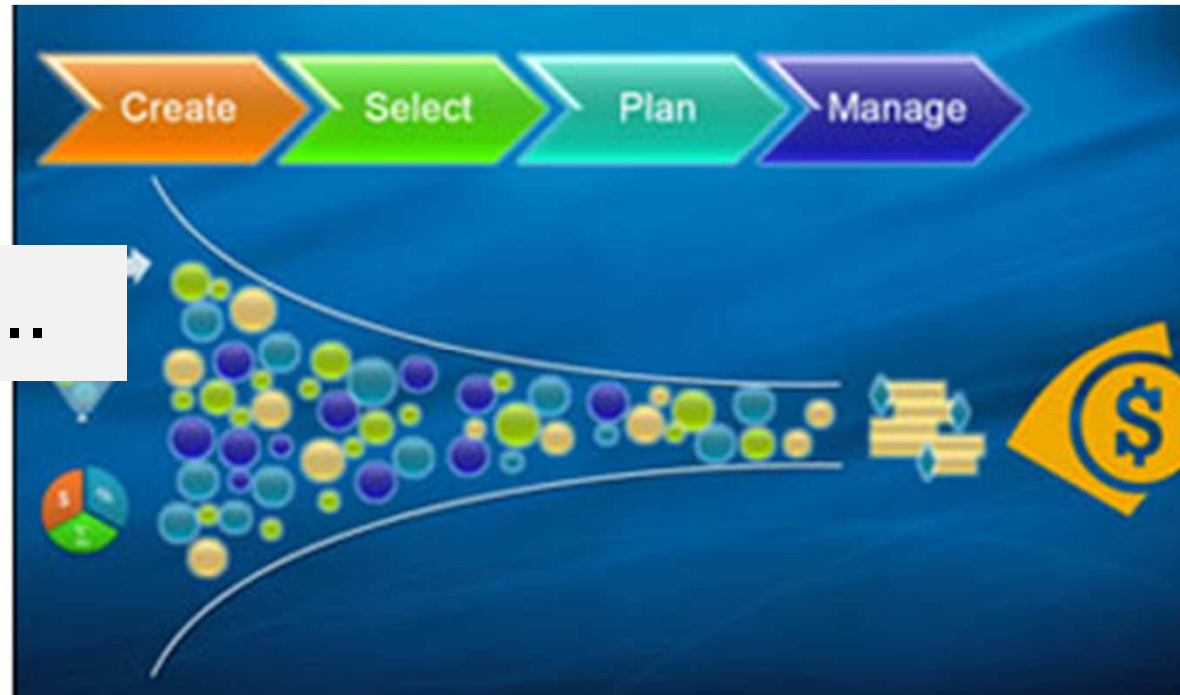
So accept to let some problems aside ...
“Don’t invest where no one is scandalized”

5 field experiments

- Let's (try to) **free up energies**
- Bank branch “**bubble**”
- 2M€ fixed price **Agile** project
- **PDCA** at the Customer Service
- **devOps** reorganization

Bank branch “bubble”

The plan ..



.. OR the best idea from the field?



« About 4 times a week, I can't answer customer claims about refused direct debit, it is not displayed in the system ... »

Fix the OR dilemma : plans **AND** ...

- Deal a **budget** of 500 men.day directly with the Sales Executive
 - specification = "improving user efficiency for the customer benefit. dot"
- Operational people own the backlog & priorities : **no ex-ante approval process**
- Demands are **delivered every month** (agile)
- Introduce a **culture of value** for the backlog
 - How often does it occur? How many people are impacted? What's the benefit they'll claim (minutes, quality/security, opportunities)? What if we don't do it?
 - E.g. : the right debit information in case of a customer claim (2000p * 5' * 4 * 50 weeks = 4000 "man days")

5 field experiments

- Let's (try to) **free up energies**
- Bank branch “**bubble**”
- 2M€ fixed price **Agile** project
- **PDCA** at the Customer Service
- **devOps** reorganization

Call Center system reengineering



1 year $\frac{1}{2}$, still in contract negotiation, 2M€





200 pages of specifications are not a contract but an education tool for the team

Bind the contract to the Why? not to the How?

10 lines contract : when asked, 180 people at the Call Center say “it is at least 50% better than before*”, release monthly, paid by user deployed, target price at target delay (18 months), supplier bears the extra cost, Bred bearing the extra delay.

* For each set of features, vote either *better* (-1), *the same* (0), *worse* (+1)

Annex 1 (not contractual) : specifications

5 field experiments

- Let's (try to) **free up energies**
- Bank branch “**bubble**”
- 2M€ fixed price **Agile** project
- **PDCA** at the Customer Service
- **devOps** reorganization

PDCA at the Customer Service

120 000 calls per year

60 000 cases (80% incidents)

No feedback loops





A new motto/goal :

***“A good customer service
is a service you don't need to call ”***

It's ok to spend time on improvements, managers support it and CIO animates the **PDCA workshops every Monday**, a backlog is opened at the Development Department



Fights against solutions thrown
before **root cause** is identified
through disciplined 5 why?

RESULTS

OBJECTIVES

PDCAs

PARETO OF
CAUSES

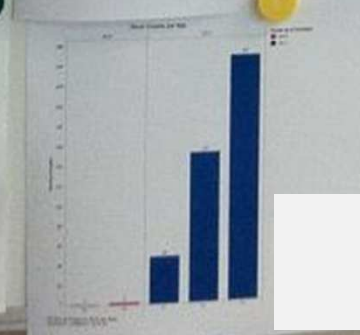
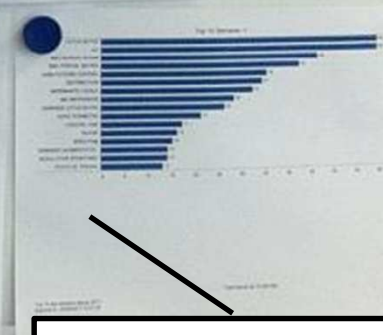
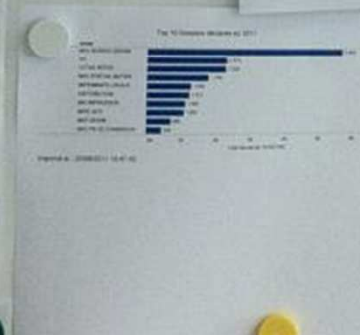
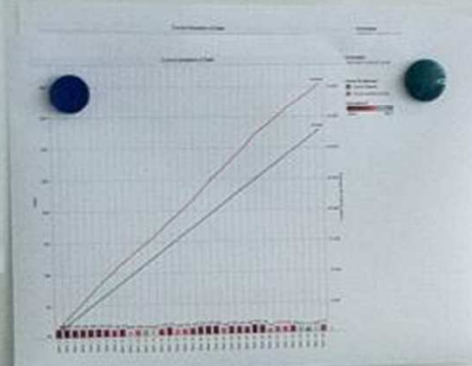
Visual Management

2011
70%
Satisfaction
Clients

2011
< 50 000 dossiers
Soit
- 700 Dossiers/mois

2011
< 100 000 Appels
Soit
- 2 000 App/mois

2011
Stock
< 300 dossiers
Soit - 40%
Pas de dossiers > 9 mois



Handwritten notes on a sticky note, part of the PDCAs section.

Handwritten notes on a sticky note, part of the PDCAs section.

Handwritten notes on a sticky note, part of the PDCAs section.

Handwritten notes on a sticky note, part of the PDCAs section.

BRED Mail - get pdca log - pierre.pe... x PDCA Etudes-Production - Canal... x

https://canalwiki.production.local/processus/pdca-etudes-production

Cette page récapitule les  PDCA mis en place dans le cadre du rituel "amélioration build-run". Chaque petit-pas porte son statut :

- **DO** : en cours de réalisation
- **CHECK** : en cours de vérification de la prédiction
- **ACT** : la solution est désormais le nouveau standard

La participation est ouverte à tous, les réunions se déroulent en salle Comète, le mercredi de 14h30 à 16h.

Problème	Cause Profonde
La sté SODIFRANCE doit réaliser des tests de performance et de non-régression dans le cadre de la migration DL/1. Ni la plate-forme de recette, ni celle de développement ne permet de les réaliser.	La plate-forme de recette est difficilement perturbable car utilisée par de nombreux utilisateurs. Elle utilise des données glissantes (miroir de la production) qui inhibe toute possibilité de automatisés.
Solution	Prédiction
Une plate-forme d'intégration avec données réinitialisables (en date figée) est mise à disposition de l'équipe modernisation, qui l'utilise en toute autonomie. Cette plate-forme a vocation à devenir multi-produit, ce qui fera l'objet de futures itérations.	La plate-forme est opérationnelle avant le 30 janvier. Sodifrance y teste régulièrement les migrations et les compare aux anciens. Sodifrance livre des tests automatisés rejouables.

Problème	Cause Profonde
Des changements risqués (systèmes, réseaux, applicatifs...) sont effectués sans communication à l'Exploitation	L'information est parfois perdue entre les ETUDES et la PROD.
Solution	Prédiction
Dans PLANIF ajouter une donnée "Risque d'incident majeur" (oui / non) MAJ le 16/12/10 par JM :	Aucun changement risqué sans comité de gestion des changements au courant

- **8000 users calls per year** because they can now reset a locked session by themselves
- **hundreds of customers calls** because we replaced generic error messages in the web site

...

5 field experiments

- Let's (try to) **free up energies**
- Bank branch “**bubble**”
- 2M€ fixed price **Agile** project
- **PDCA** at the Customer Service
- **devOps** reorganization

Development teams
often pollute ...



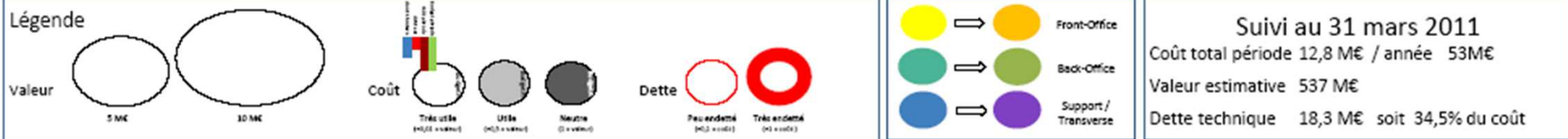
.. **costs** or **incidents** at the Operations
.. **calls** at the Customer Service

and Support
Functions add **delays**



It's cool to be a bottleneck

architecture, security, purchasing, technical expertise, methods ..

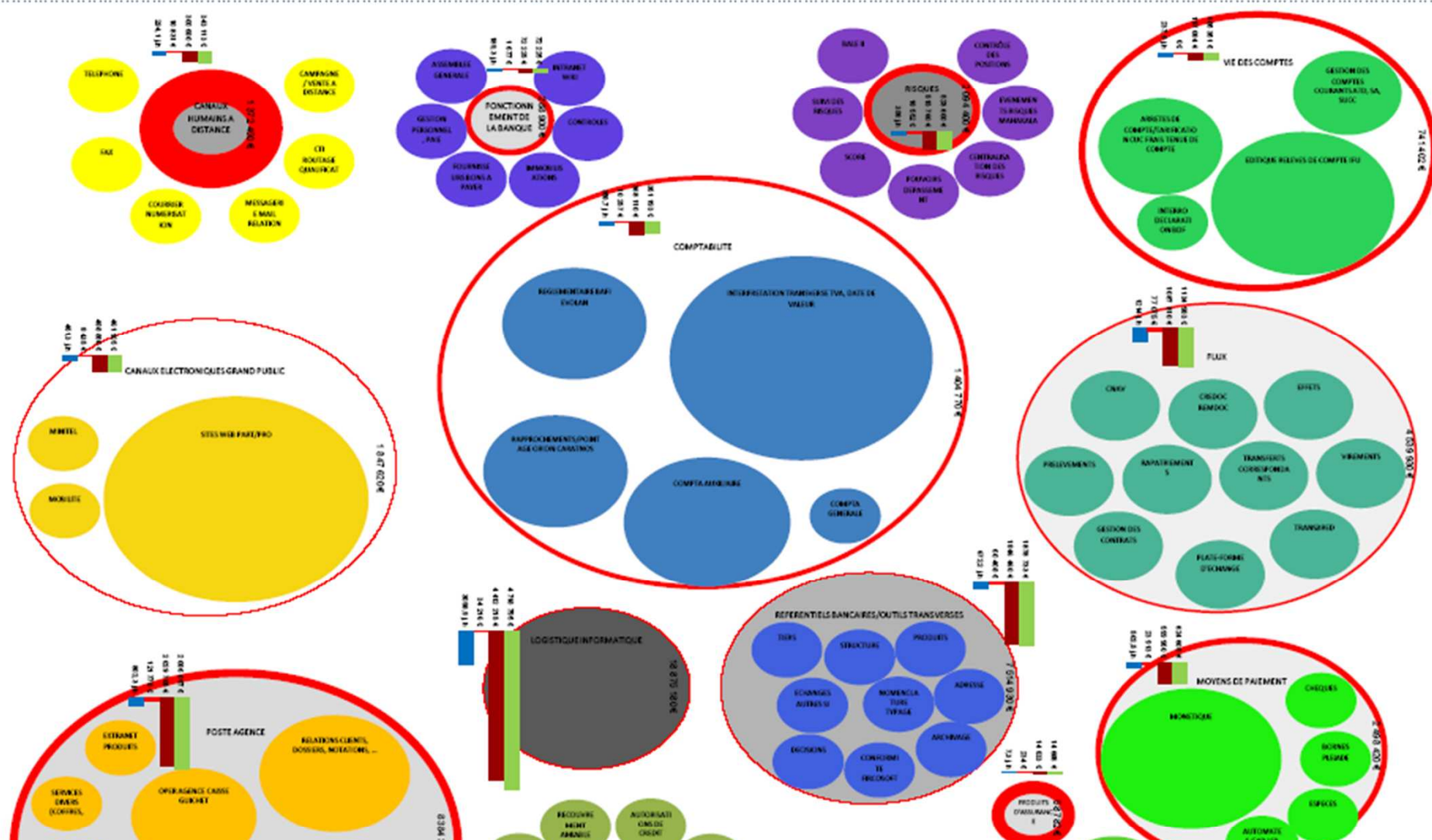


C
A
R
T
O
G
R
A
P
H
I
E

D
U

S
I

B
R
E
D



Why split responsibilities ? You build it, you run it, and you bear all the costs (actual and future, aka technical debt)

devOps : no silver bullet a cultural shift



Think as if you had to **sell your product outside of the company**

An integrated banking distribution system
A (Cobol/DB2/JBoss) “appEngine”



Make it so **easy to use that people don't call you**

“dilute your job, you're promoted” vs “it's cool to be a bottleneck”

Promote **pair-managing** : choose your “opposite”

Nobody is perfect and always available

Final bottleneck : **TRUST**

If you want continuous improvement,
you will have to **increase cooperation**,
i.e. increase the “**shared territories**”



To do so, you will need that the people :

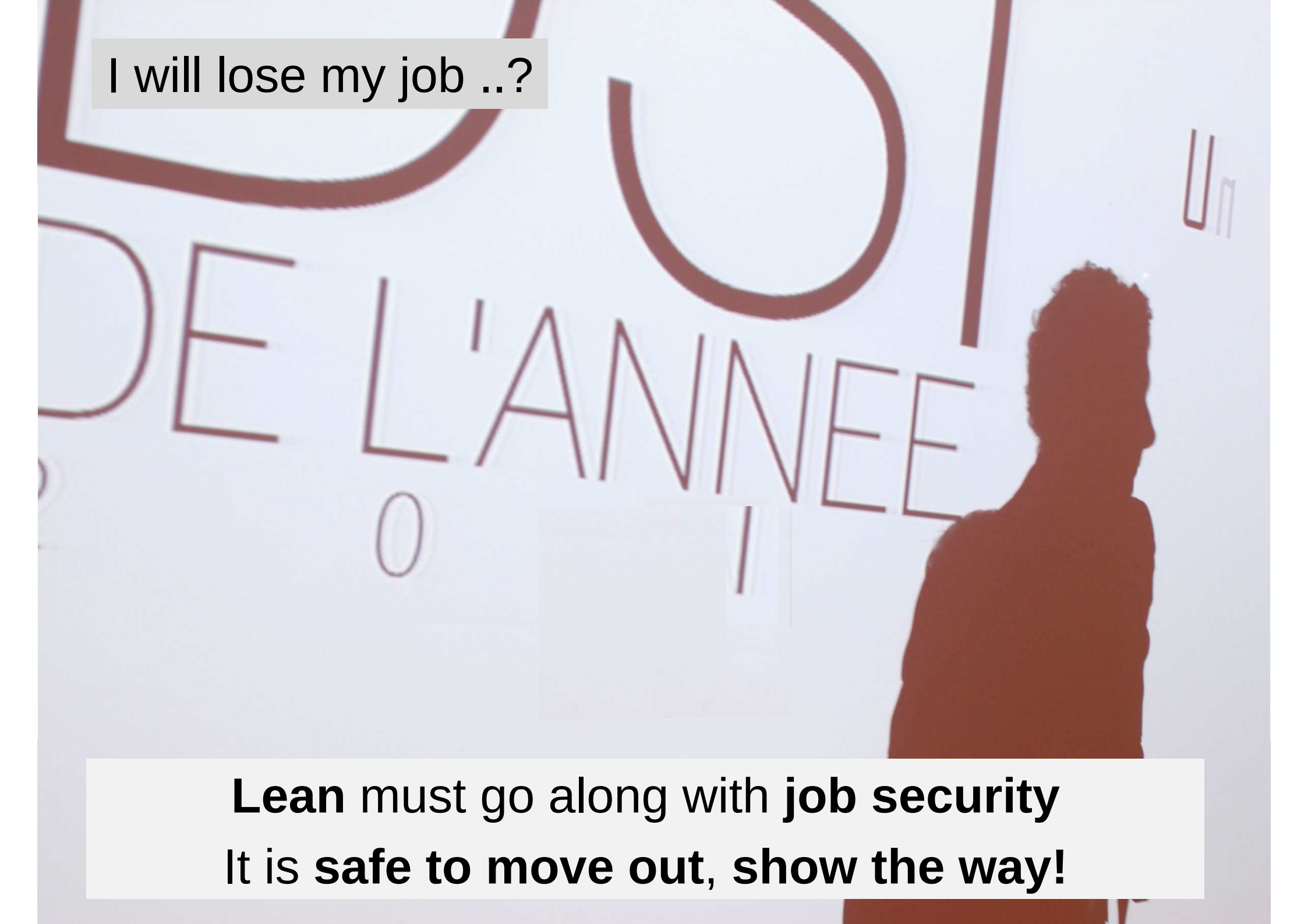
release power (if they have some)

show their **vulnerabilities** (they must have some)

If I'm no longer
essential, I may
loose my job ..

they gonna realize I code
more slowly than them, I
will **loose my job**

How many people are ready to?

A person's silhouette is visible on the right side of the image, looking towards a large screen. The screen displays the text 'DE L'ANNEE' in a large, outlined font, with a large '0' below it. The background is a light blue gradient.

I will lose my job ..?

Lean must go along with **job security**
It is **safe to move out, show the way!**

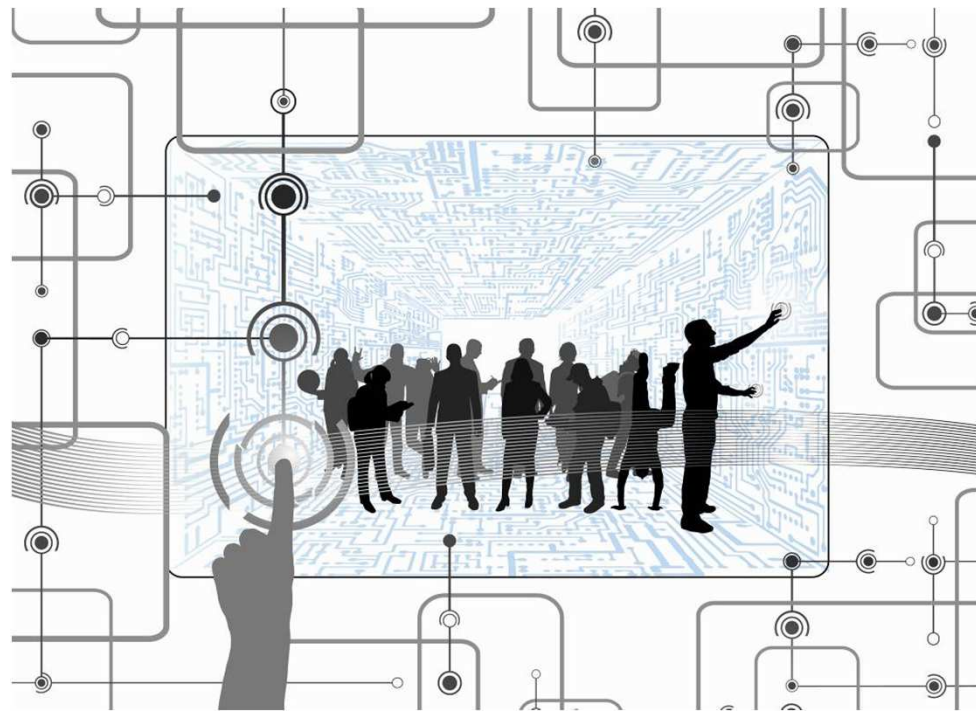


A lean startup
team=product **goal** is usefulness, profit a **constraint**

Continuous improvement
As easy to use as to contribute to
Dilute your job, you're promoted

Convivial Computing

twitter @ppezziardi



<http://informatique-conviviale.eyrolles.com/>