

Cours 4

Développement agile

Management Agile, Métriques et vélocité
Conclusion sur les Méthodes Agiles



Sommaire

Management du développement agile

- Piloter grâce au BurnDown chart - 3
- Piloter avec la vélocité - 16
- Conclusion sur l'Agilité - 24
- Je retiens - 29

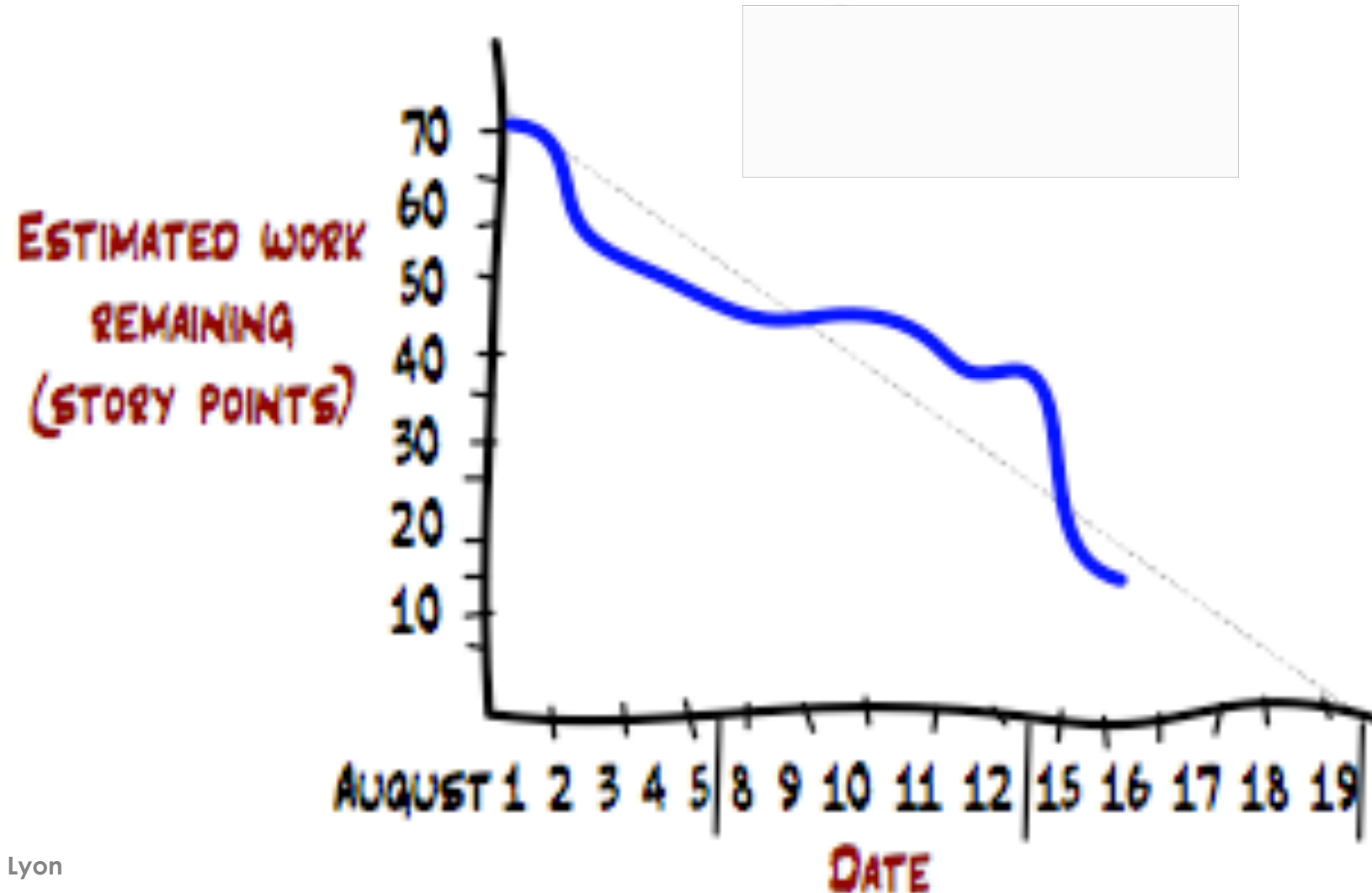


Le pilotage

Savoir où on en est et où on va en agile

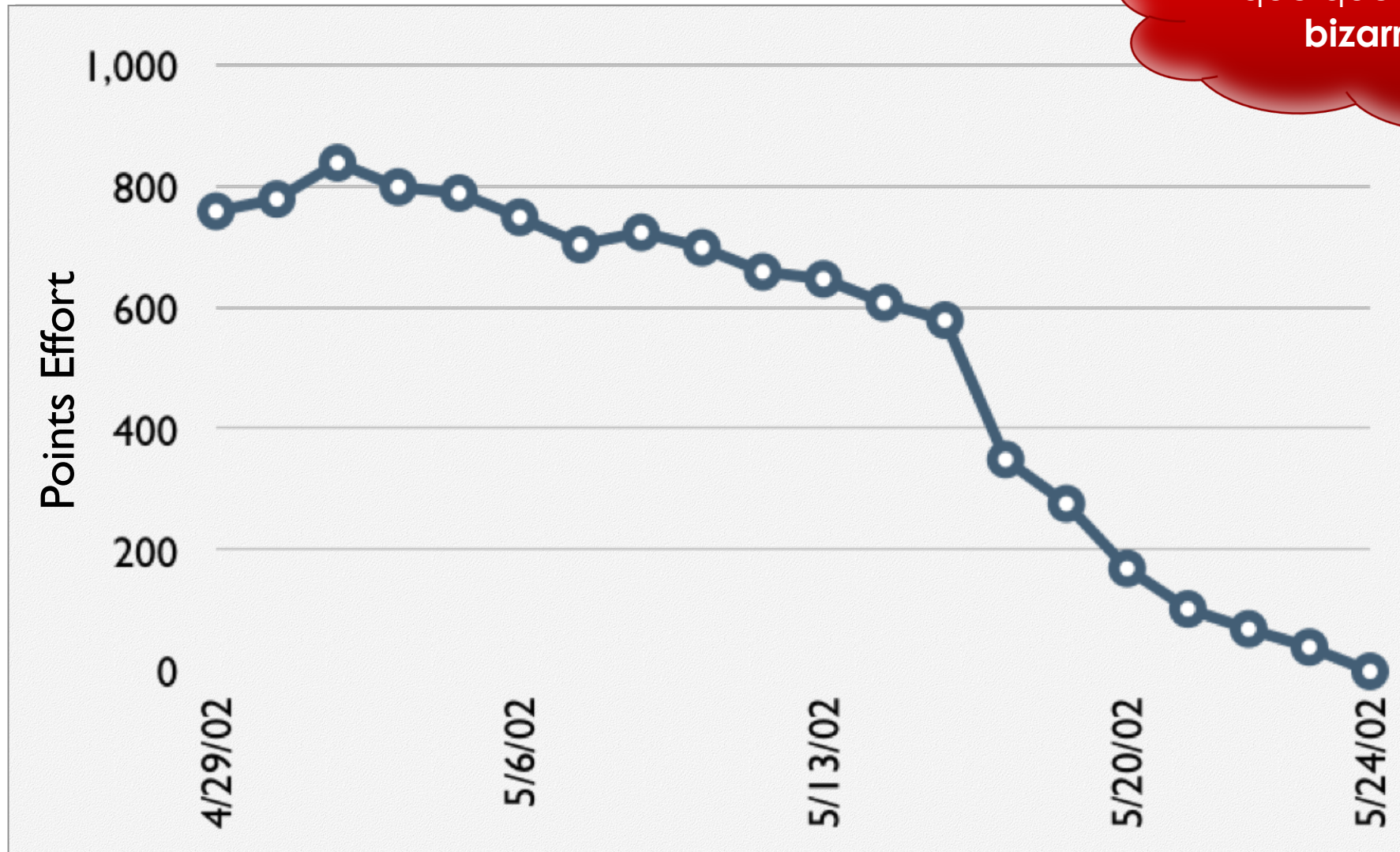
Burndown Chart de Sprint

Ce qui a été fait et ce qui reste à faire



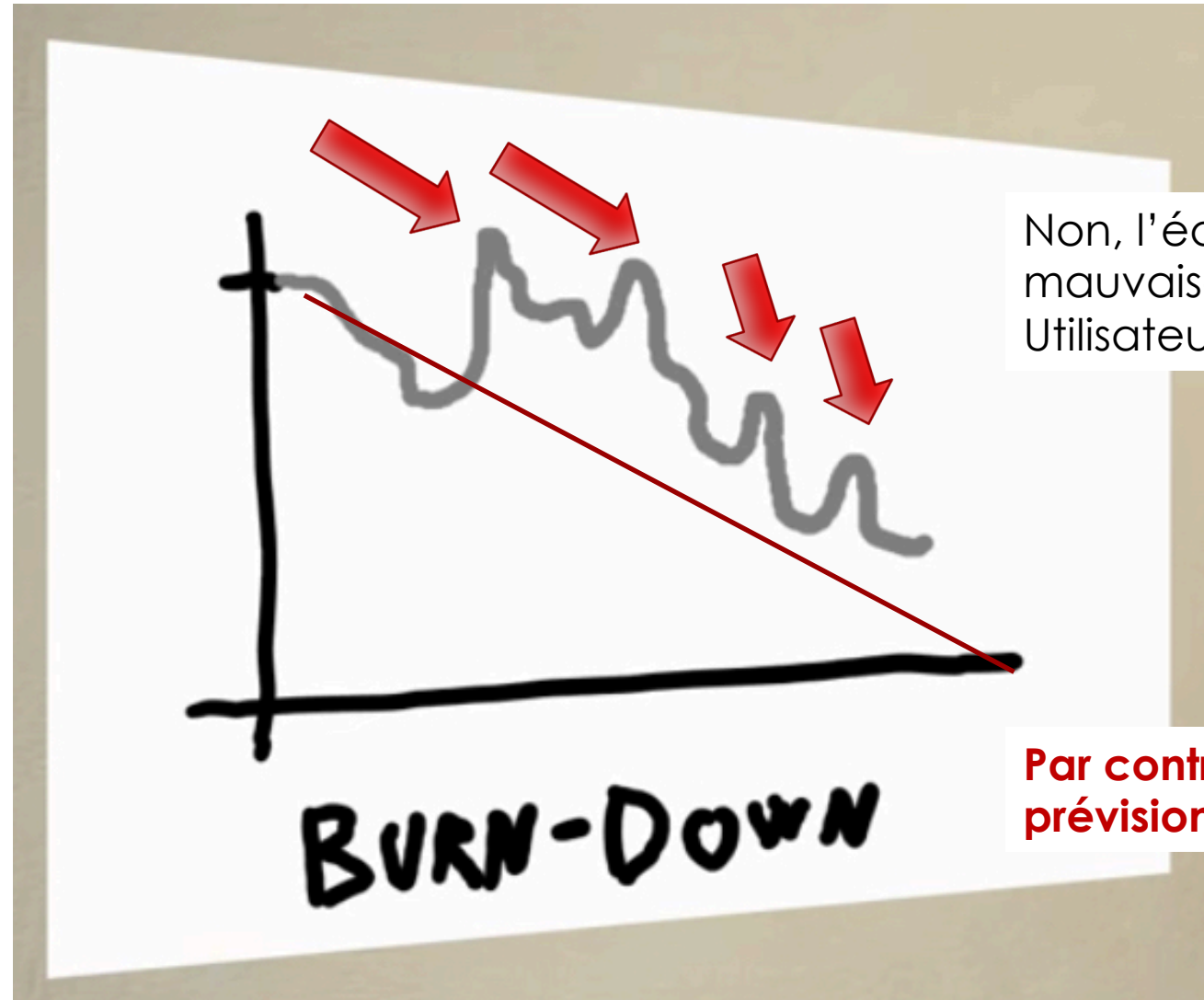
Un burndown de sprint

N'y a-t-il pas
quelque chose de
bizarre ici ?



6

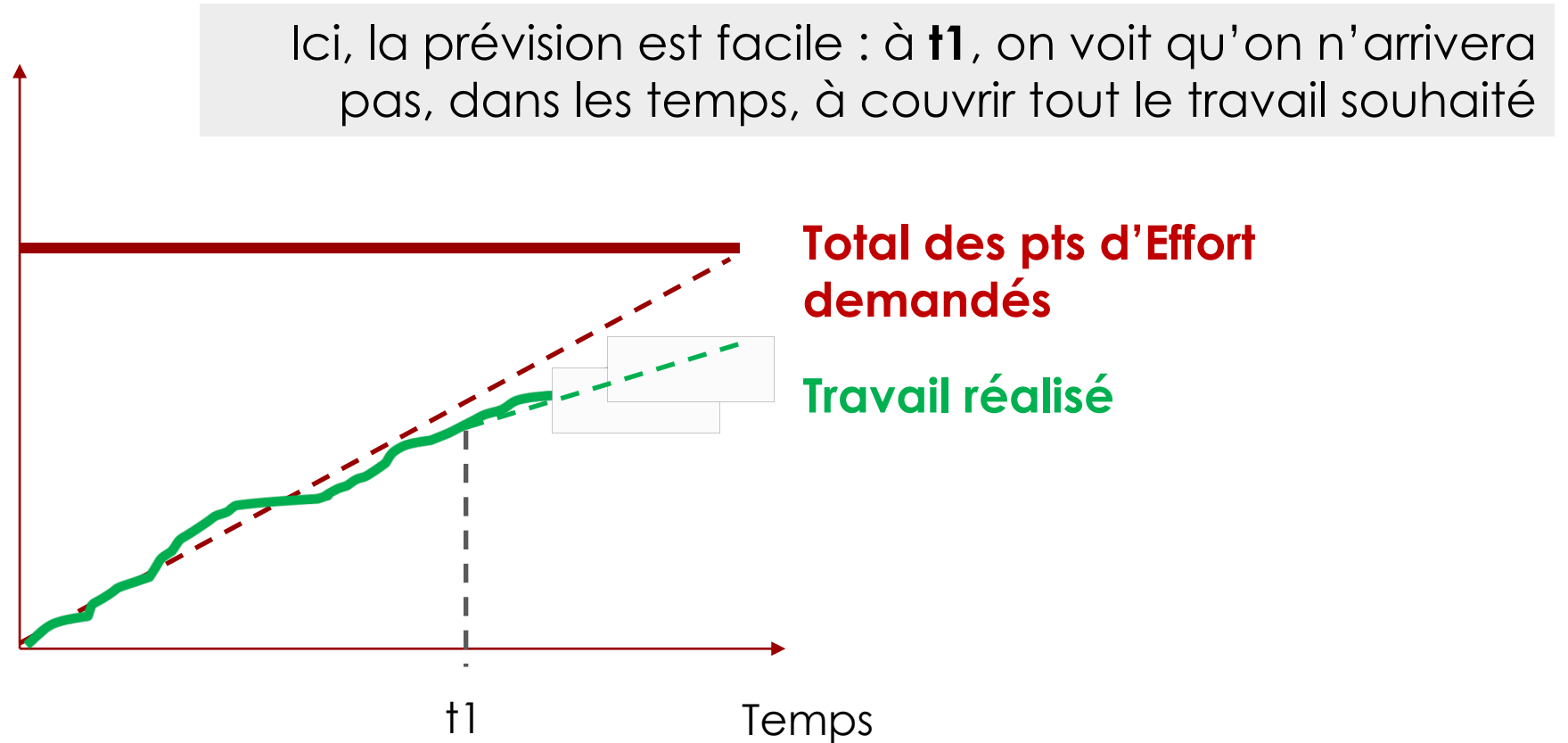
Que penser de ce burndown chart ?
Est-il mauvais ?



Analysons ce BurnUp Chart

Avec un périmètre fixe :

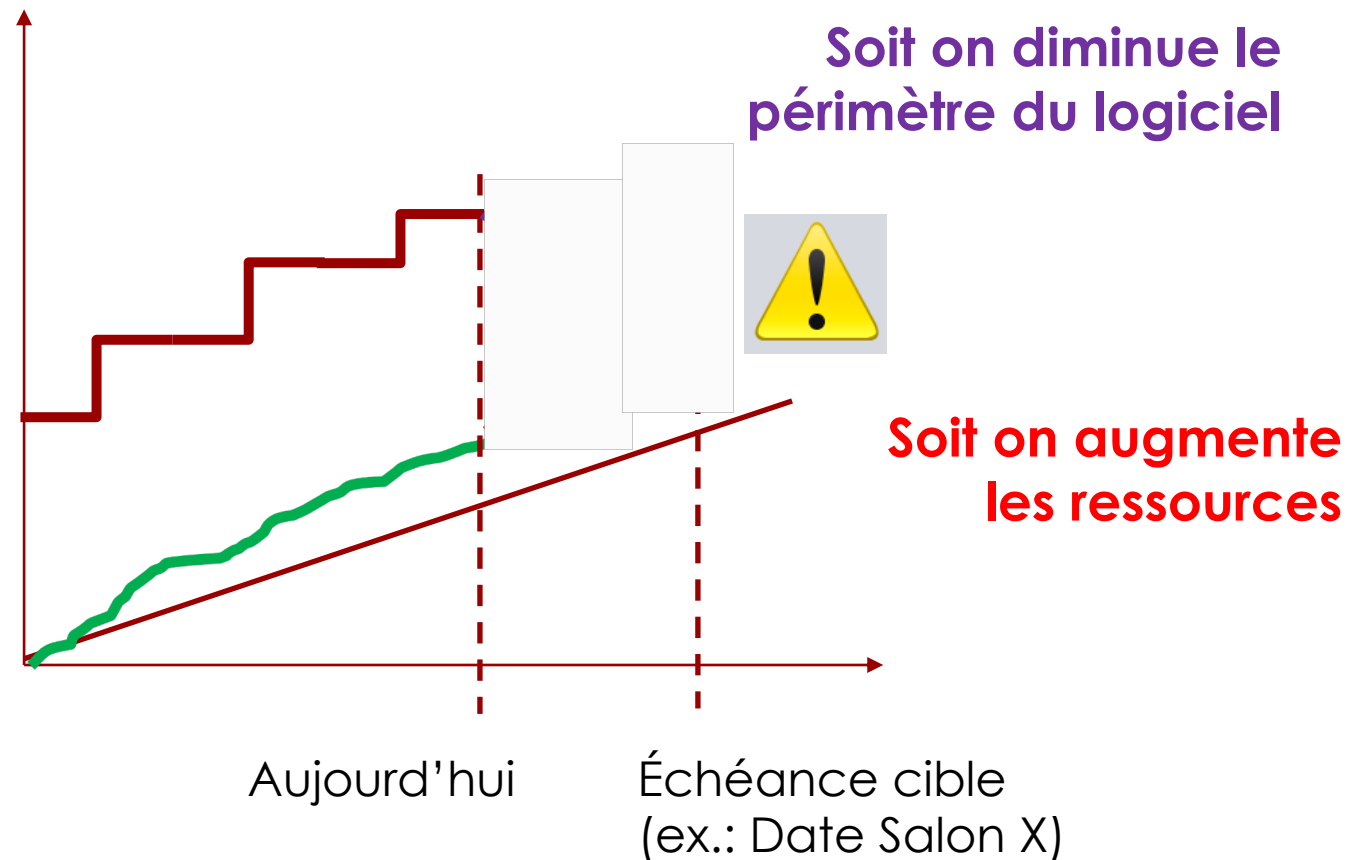
Points
d'effort



Analysons ce BurnUp Chart (2)

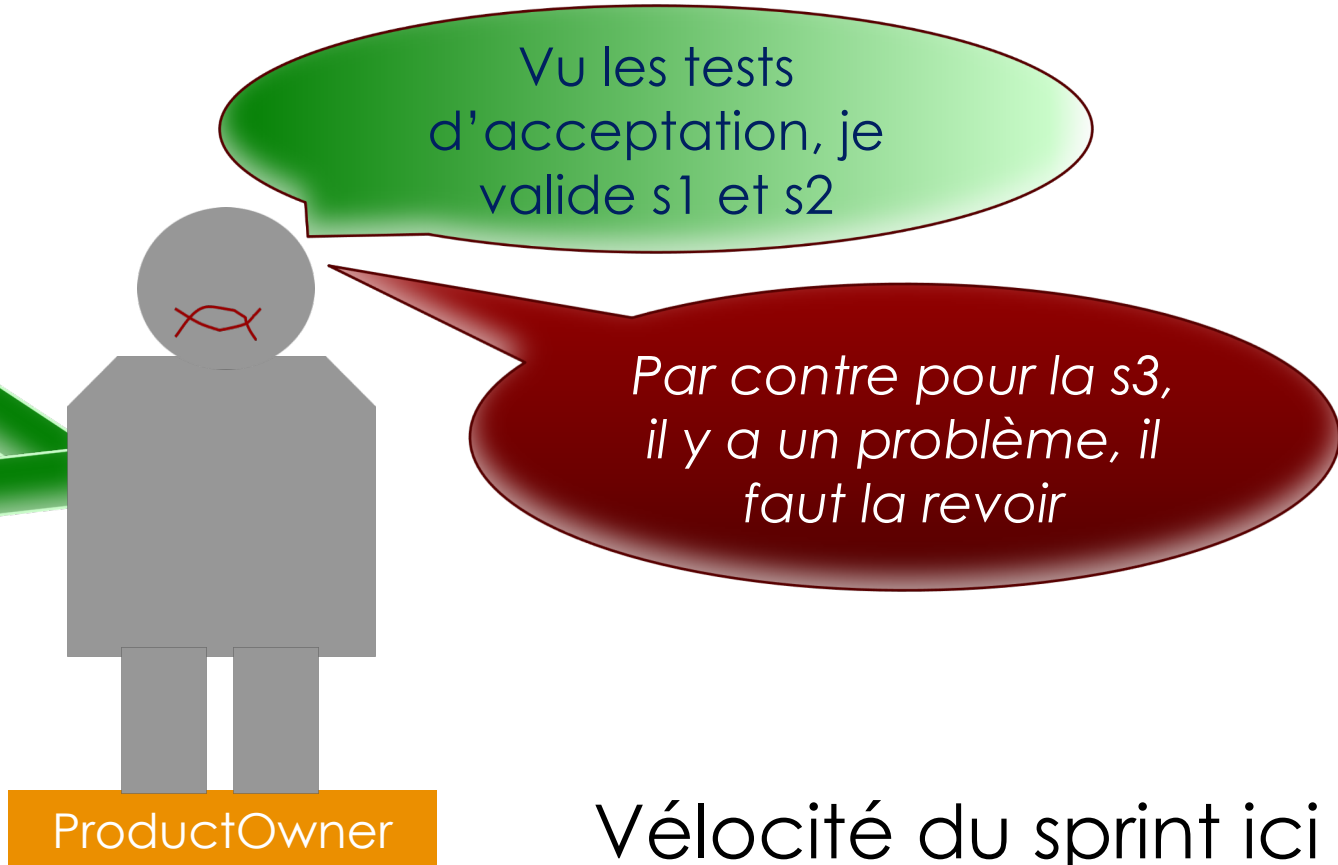
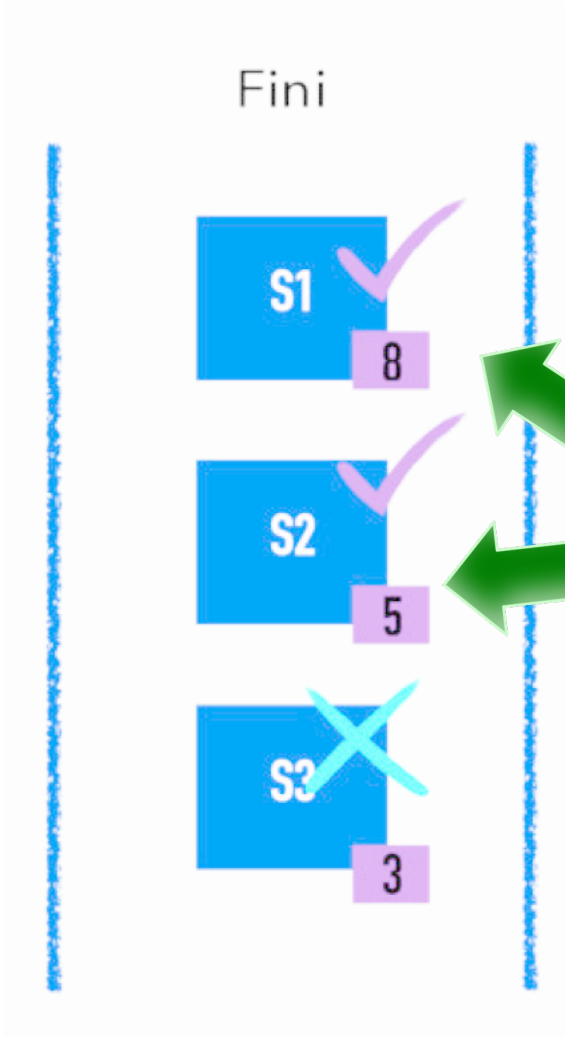
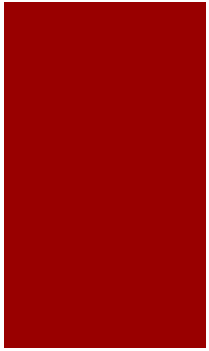
Avec un périmètre variable :

à **t1**, si on décide que le BKLog est stable, on voit qu'on n'arrivera pas, dans les temps, à répondre au travail demandé



Calcul de la vélocité

Somme de points d'effort
des US VALIDÉES
par le Product Owner



Vélocité du sprint ici :
 $8 + 5 = \mathbf{13}$

Métrique : capacité

- DEF Nombre de points (effort estimé) par sprint que l'équipe **peut** réaliser
 - Définie par la **vélocité** du sprint précédent
- Comparaison capacité / vélocité = aptitude de l'équipe à **prévoir ses engagements**

- La vélocité est réévaluée à chaque itération
 - Au début aléatoire, elle s'affine ensuite et se stabilise.

L'estimation de l'effort peut évoluer

- Au fur et à mesure des réalisations, l'équipe est mieux armée pour évaluer les efforts des user stories / des tâches

- Un exemple de backlog :

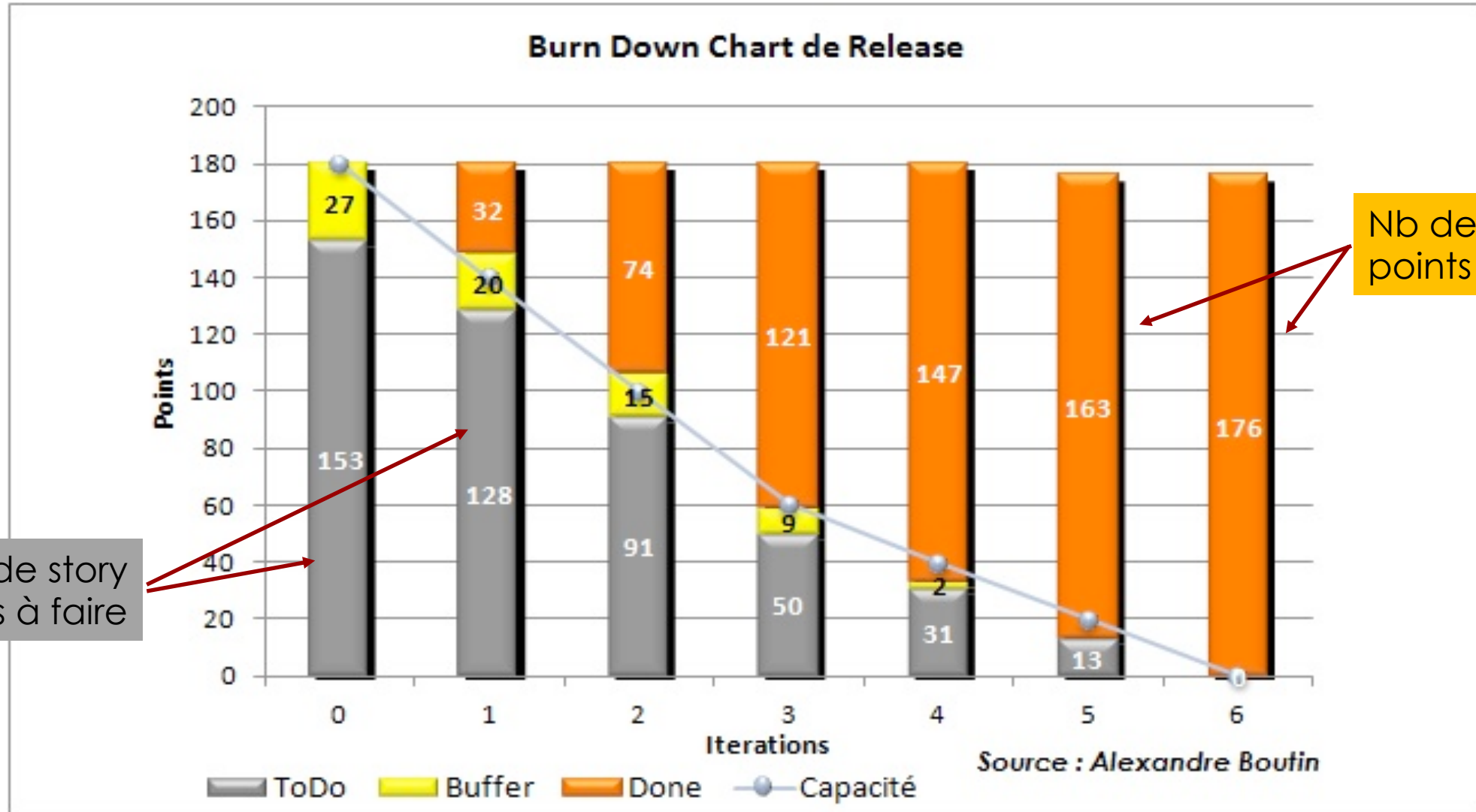
Tâches techniques



Priorité	Element	Détails (URL wiki)	Effort initial estimé	Nouvel effort estimé					
				Reste à faire sur le Sprint					
				1	2	3	4	5	6
1	En tant qu'acheteur, je veux placer un livre dans un panier de courses (cf croquis de l'IHM sur le wiki)	...	5						
2	En tant qu'acheteur, je veux supprimer un livre de mon panier	...	2						
3	Améliorer les performances des transactions (cf performances cibles sur le wiki)	...	13						
4	Investiguer une solution pour accélérer la validation des cartes de crédit (cf métriques de performance sur le wiki)	...	20						
5	Mise à jour des serveurs avec Apache 2.2.3	...	13						
6	Diagnostiquer et réparer l'erreur de calcul de commande (ID bugzilla 14923)	...	3						
7	En tant qu'acheteur, je veux créer et sauvegarder une liste d'envies	...	40						
8	En tant qu'acheteur, je veux ajouter ou supprimer des articles dans ma liste d'envies	...	20						

Un ex. de pilotage à l'aide de la vélocité

(ici rempli à la fin)

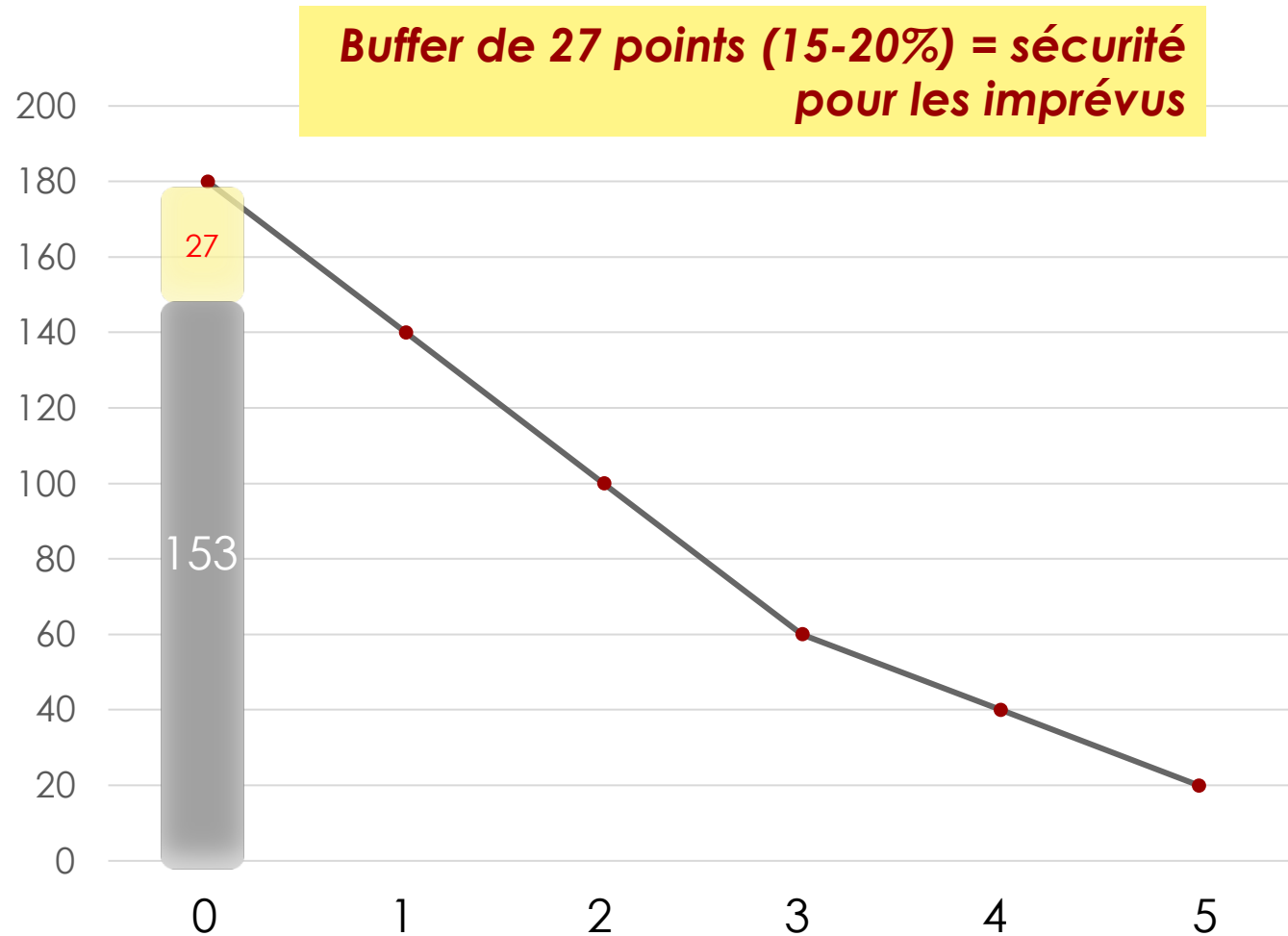


Nb de story points à faire

Nb de story points effectués

Interprétation

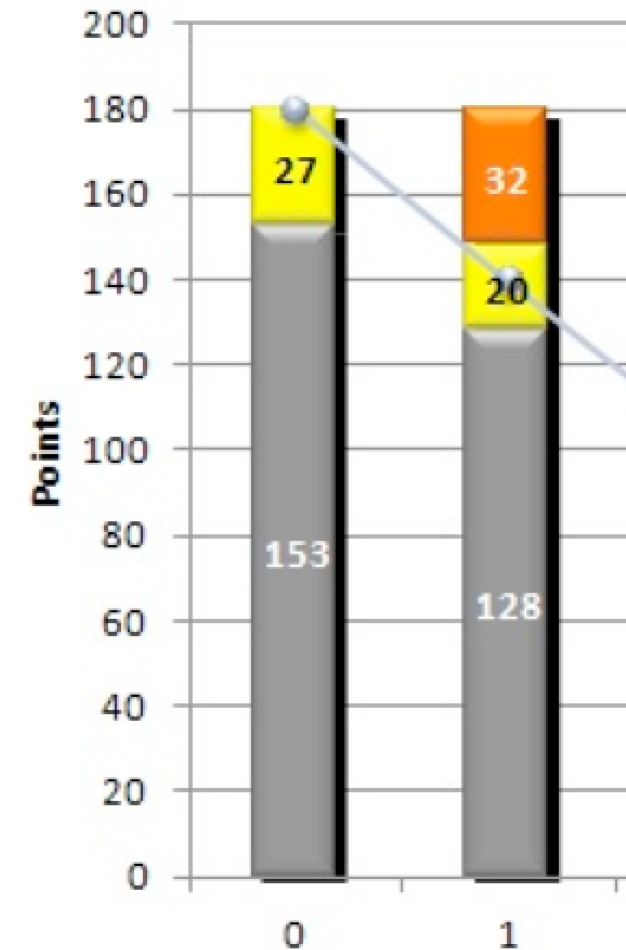
- La capacité de l'équipe est connue : elle est fixée à 40 pts
 - Après la 3^{ème} itération, l'équipe sera diminuée : on passera à 20 pts de capacité (c'est prévu)
- Il y aura 6 itérations : sur ce cycle on pourra ainsi produire $3 \times 40 + 3 \times 20 = 180$ pts d'efforts
- Fin du Sprint 0 : les stories du Backlog sont estimées à **153 points**.



Itération 1

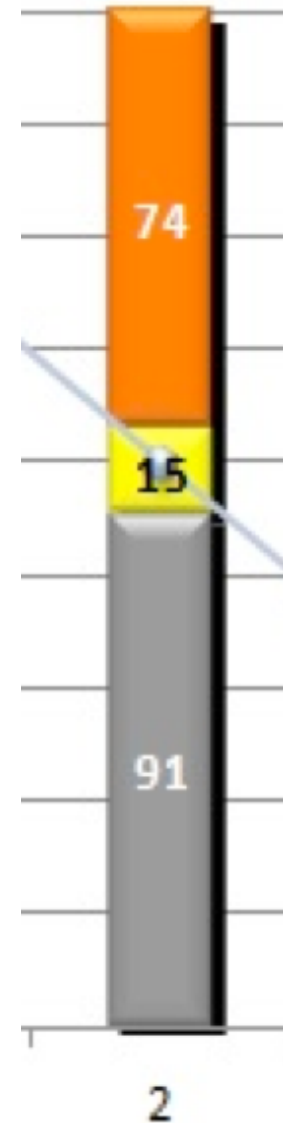
L'indicateur de capacité restante (140) est dans la zone jaune (le buffer) :

- On a moins réalisé que prévu
 - (32 au lieu de 40)
- On voit qu'il ne sera pas possible, avec la capacité prévisionnelle, de délivrer le backlog actuel
 - 128 + le buffer (20)



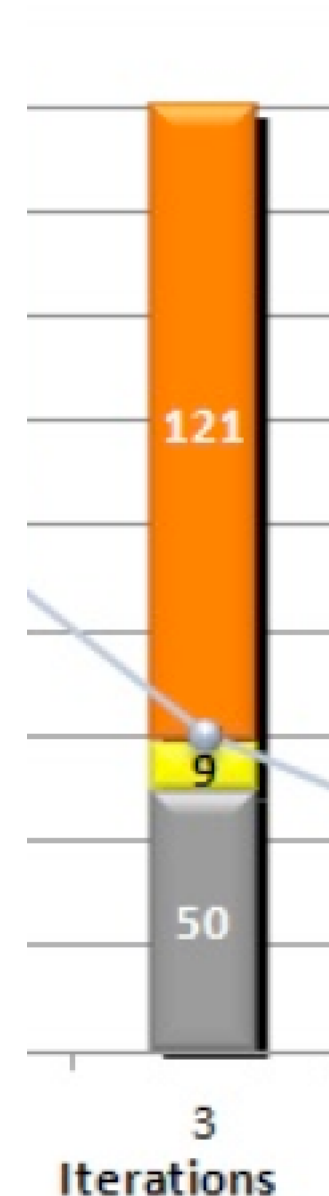
Itération 2

- On a réalisé 42 pts de US (74 – 32)
- Mais on n'est toujours pas dans le **gris**
 - Le *restant à faire*
- **Capacité** < au reste à faire (avec le buffer)...
 - Risque de ne pas réaliser tout le backlog dans les temps
- Prévoir soit de négocier un plus petit périmètre, soit augmenter l'équipe, soit booster les codeurs !



Itération3

- L'équipe réalise une **vélocité de 47** (pour 40 attendu)
- L'indicateur de capacité restante (60) est dans le orange ... tout va bien : le RAF est égal à la capacité de l'équipe
- L'équipe **devrait pouvoir** implémenter tout le backlog + le buffer des imprévus.

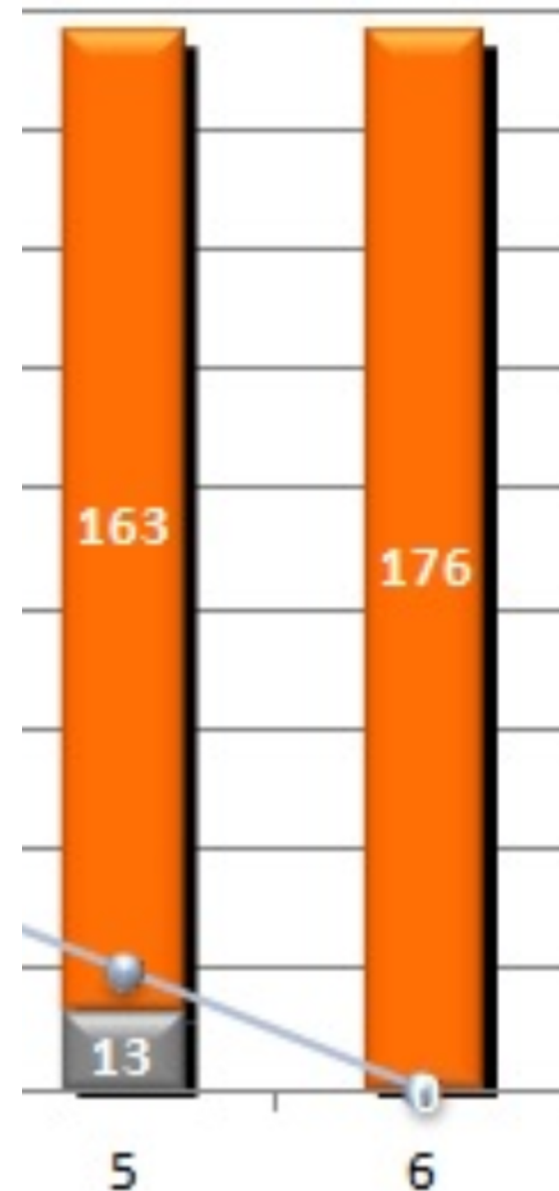


(...) Itération 5

- Le Product Owner décide de **sécuriser sa Release** :
 - Il ne met que 13 points de stories à faire pour la 6^{ème} et dernière itération
 - (Au lieu des 20 prévus)
- *Une très bonne pratique !*

Itération 6 : la Release est terminée dans les temps

- L'équipe peut en profiter pour faire quelques *stories techniques*
 - À intégrer dans la Release suivante



Principe de la « Météo d'hier »

- En Scrum, la **capacité** de l'équipe sur un sprint est définie par la **vélocité** du sprint précédent
 - exprimée en *story points*
- Ainsi on sait définir la **taille du backlog de sprint** :
 - Si on a une vélocité de 30 points au sprint#3, au Planning Game du sprint#4, on choisit les items prioritaires du backlog représentant **30 pts.**
- C'est une estimation de la **charge de travail optimale**
 - = le '**WIP Limite**' en Kanban
- **Repose sur l'hypothèse** que l'équipe va avoir la même productivité, d'une itération à l'autre
 - Ce qui est le cas après quelques sprints en général

Le rôle du ScrumMaster

19

- **Garant du cadre méthodologique Scrum :**
 - Forme les membres aux pratiques agiles
 - Anime les différents « rituels » de Scrum
 - **“Coach” agile**
 - S’assurer de la bonne compréhension entre le Product Owner et l’équipe
 - Sert **d’interface** entre l’équipe et le monde extérieur; **chien de berger (protecteur)**
 - **N’a pas vocation à diriger, mais à guider l’équipe**
 - Ne décide pas du rôle de chacun durant **les sprints**
 - Aider chacun à franchir les différents obstacles rencontrés
 - Plus l’équipe gagnera en expérience et en autonomie, plus le rôle du Scrum Master se réduira
- **Possède de fortes compétences humaines** : diplomatie, empathie, pédagogie, humilité; analogie : **Maître de Jeu dans les jeux de rôle**

Conclusion

Sur les Méthodes Agiles

Une maturité acquise

21

- Les outils associés sont maintenant disponibles sur le marché
 - y compris en Open Source
- Les formations, certifications, conférences, livres, blogs se sont multipliés
- Prise de position **en faveur de l'approche Agile** de la part des organismes faisant "autorité" en matière de gestion de projet informatique
 - En 2012, le Gartner Group (cabinet US d'observation des sociétés d'IT - Information & Technology) invite à [abandonner le Cycle en V](#)

Contre-indications

- **Y a-t-il des contre-indications à l'agilité ?**
Oui, ce n'est pas une méthode universelle.
- L'agilité n'est recommandée que pour développer des produits et/ou des services à **forte dose d'innovation** avec un minimum de professionnels du domaine (dév connaissant le Métier).
- Quand il n'a pas de **Product Owner** : inutile d'imaginer être agile
- **Les méthodes classiques** avec leurs processus et procédures prédéfinis **sont préférables** :
 - Pour créer de *petites variantes* de produits existants
 - Si la majorité des développeurs a peu d'expérience opérationnelle de projet
- Lire aussi (Aout 2015) <http://www.morisseauconsulting.com/2015/08/24/quand-scrum-ne-marche-pas/>

Pourquoi parfois ça ne marche pas ?

23

- La « **solution miracle** »
 - Certaines équipes rencontrent des difficultés concrètes, une faible productivité ou peu de qualité, et elles espèrent que l'Agile va faire **miraculeusement** remonter ces indicateurs.
 - Mais l'Agile est une **approche systémique**, au service d'objectifs de hauts niveau tels que '*S'auto organiser*', '*Satisfaire les clients en répondant à leurs besoins changeants*'
 - Il faut *un travail de fonds* pour améliorer une organisation en difficultés.
- Ressenti de SCRUM = **lourdeur** : les *Daily meetings*, *Planning Games*, *rétrospectives*. Parfois vécu comme trop contraignant.
 - Souvent adapté à la culture d'entreprise
- **Sans soutien managérial et technique** significatif, l'agilité ne survit pas.

Retour de 15 ans d'expériences

Pablo Perno

<https://www.areyouagile.com/2013/11/agile-a-grande-echelle-cest-clair-comme-du-cristal/>

- C'est dur, **Scrum** : cela demande un investissement, et cela renouvèle les rôles, les responsabilités, l'organisation et la culture de la compagnie.
- Tout le monde **n'est pas ravi** et tout le monde **ne réussit pas**.
- **Si on n'a pas envie** :
 - *D'apprendre, de faire confiance, d'accepter l'incertitude...*
 - *De remettre l'humain au centre des décisions...*

On n'aimera pas l'agilité.

- De nombreuses entreprises souhaitent malgré tout être agiles et se tournent vers **Kanban**
 - Kanban permet de se *dire agile* sans imposer toutes les contraintes de Scrum.
- Un département de 50 personnes avec **5 à 7 équipes** auto-organisées de 7 personnes, travaillant sur **un même sujet** est tout à fait envisageable.

Je Retiens

25

- On a vu comment les métriques comme le **burndown chart** ou la **vélocité** permettent de manager le projet en minimisant les risques d'échec du projet
- Le rôle du **ScrumMaster**
- L'agilité est d'abord un **état d'esprit** :
 - Livrer tôt, retours client fréquents, proximité des utilisateurs
- C'est tout sauf la pagaille
 - Il y a des **indicateurs** de l'avancement (métriques), qui permettent de toujours savoir où on en est du projet
- On a vu surtout ici la méthode **SCRUM**
- Il existe d'autres méthodes notamment **XP** (eXtreme pROgramming) et **KANBAN**, plus tournées vers les **développeurs** (notions d'ingénierie : TDD, refactoring propre, etc).

Références supplémentaires

- Un grand nb de **ressources libres (PDF)** chez Pablo Perno :
<https://www.areyouagile.com/ressources/>
- <https://frenchsug.org/> - Scrum User Group Français
- <http://www.aubryconseil.com/> - Blog de Claude Aubry, consultant Scrum et Product Owner

étriques

- Comment faire **une rétrospective réussie**

<http://blog.octo.com/la-retrospective-dont-vous-etes-le-heros/>

- Pb humain : que faire en cas **d'incompétence** au sein de l'équipe Agile ?

<http://kelibia.eu/kel/content/que-faire-face-incompetence-sein-equipe-agile>