



# Scrum, Etat de l'art

---

François Potentier,  
10 octobre 2008

# Sommaire

---

- Introduction à Scrum
- Rappel sur les méthodes agiles
- Scrum
  - Principes clés
  - Les rôles
  - Planifier un projet
  - Organisation
  - Indicateurs de projet
  - Techniques d'ingénierie logicielle
  - Scrum pour de grandes équipes
- Perspectives
- Conclusion

# Introduction à Scrum



Source : <http://commons.wikimedia.org>

- Scrum = mêlée en rugby
  - Objectifs :
    - Satisfaire au mieux les besoins du client
    - Maximiser les chances de réussite du projet
  - Méthode itérative et incrémentielle
  - Equipes de 8 personnes. Mécanismes d'extension
  - Méthode agile la plus utilisée avec eXtreme Programming
- 
- **1986** : « The new new product development game »
  - **2001** : K. Schwaber et M. Beedle publient « Agile software development with Scrum ».

# Rappel sur les méthodes agiles

---

- Manifeste de l'agilité publié en **2001**
- **4 valeurs** :
  - Les personnes et les interactions plutôt que les outils et les processus
  - Le logiciel fonctionnel plutôt que de la documentation exhaustive
  - La collaboration avec le client plutôt que la négociation de contrat
  - L'adaptation au changement plutôt que le respect d'un plan pré-établi

# Scrum – Principes clés

---

- Conforme au manifeste de l'agilité
- Met l'accent sur :
  - Auto-organisation de l'équipe
  - Pouvoir de décision donné à l'équipe
  - Délais fixes
  - Sprint en isolement
  - Réunions quotidiennes
  - Livrer un logiciel fonctionnel - démonstration du résultat du sprint
  - Planning adaptatif

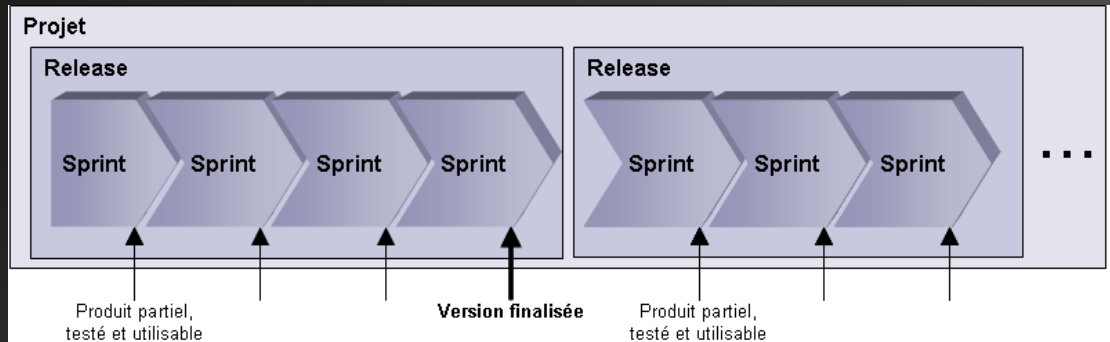
# Scrum – Les rôles

---

- Les poules et les cochons
- Les cochons :
  - Le product owner
  - Le scrummaster
  - L'équipe
- Les poules :
  - Tous ceux qui ont un intérêt dans le projet
- Certifications

# Scrum – Planifier un projet

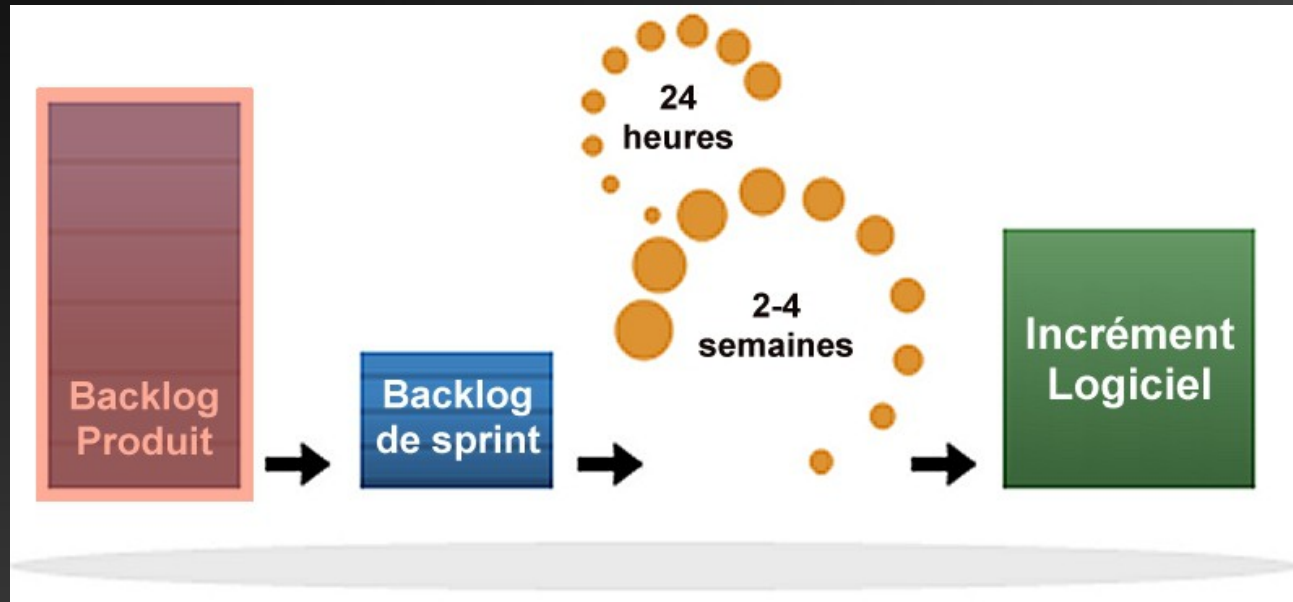
Backlog produit - Site marchand XY					
ID_Item	Titre	Importance	Estimation	Démonstration de la fonctionnalité	Commentaires
SPRINT 1					
1	Besoin 1	130	12	xxxxxxx	xxxxxxx
2	Besoin 2	120	9	xxxxxxx	xxxxxxx
3	Besoin 3	115	20	xxxxxxx	xxxxxxx
SPRINT 2					
4	Besoin 4	110	8	xxxxxxx	xxxxxxx
5	Besoin 5	100	20	xxxxxxx	xxxxxxx
6	Besoin 6	95	12	xxxxxxx	xxxxxxx
SPRINT 3					
7	Besoin 7	80	10	xxxxxxx	xxxxxxx
8	Besoin 8	70	8	xxxxxxx	xxxxxxx
9	Besoin 9	60	10	xxxxxxx	xxxxxxx
10	Besoin 10	40	14	xxxxxxx	xxxxxxx
SPRINT 4					
11	Besoin 11	35	4	xxxxxxx	xxxxxxx
12	Besoin 12	25	6	xxxxxxx	xxxxxxx
13	Besoin 13	10	7	xxxxxxx	xxxxxxx
14	Besoin 14	10	11	xxxxxxx	xxxxxxx
15	Besoin 15	10	3	xxxxxxx	xxxxxxx



Source : <http://fr.wikipedia.org>

- Constitution du **backlog produit** par le product owner.
- Répartition en **sprints** et en **releases**.

# Scrum – Organisation 1/5



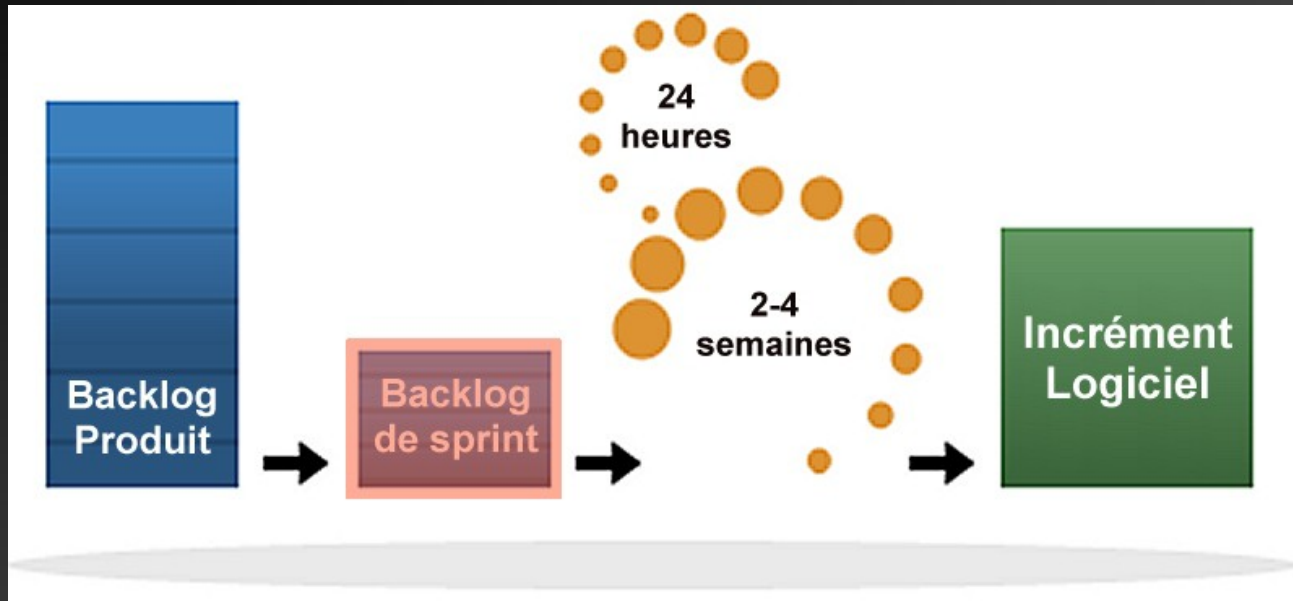
Source : [www.scrumalliance.org](http://www.scrumalliance.org)

## 1. Backlog produit (ou catalogue des besoins)

- Besoins priorisés par le product owner
- Besoins évalués par l'équipe



# Scrum – Organisation 2/5

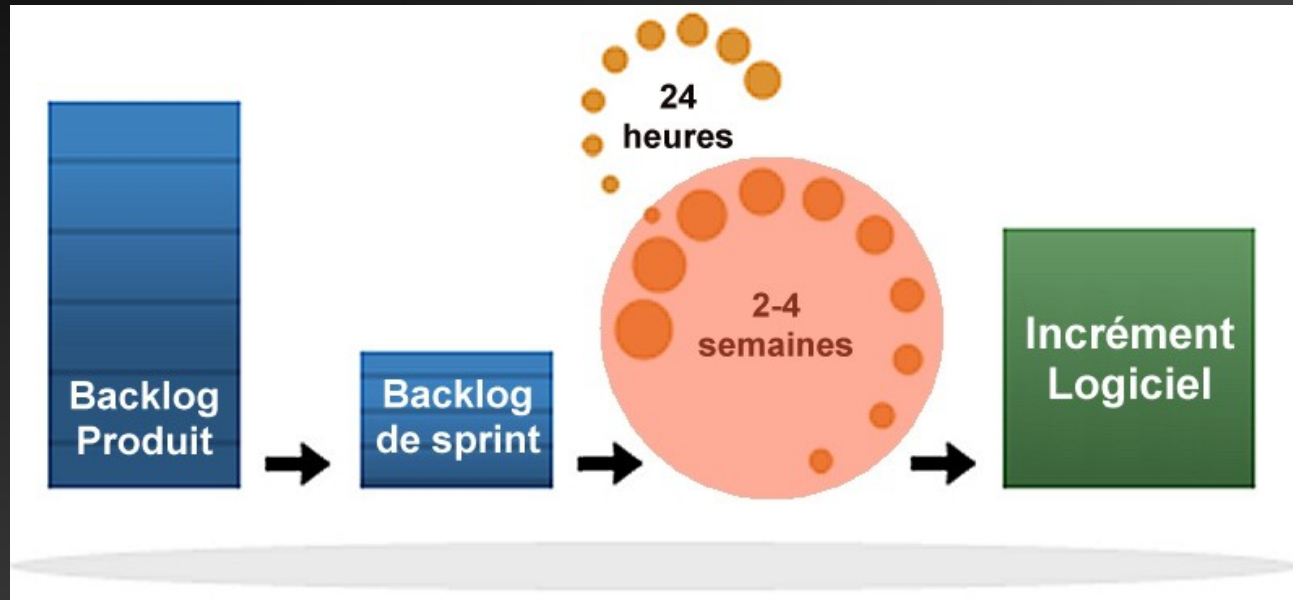


Source : [www.scrumalliance.org](http://www.scrumalliance.org)

## 2. Backlog de sprint

- Extrait du backlog produit
- Besoins éclatés en tâches

# Scrum – Organisation 3/5

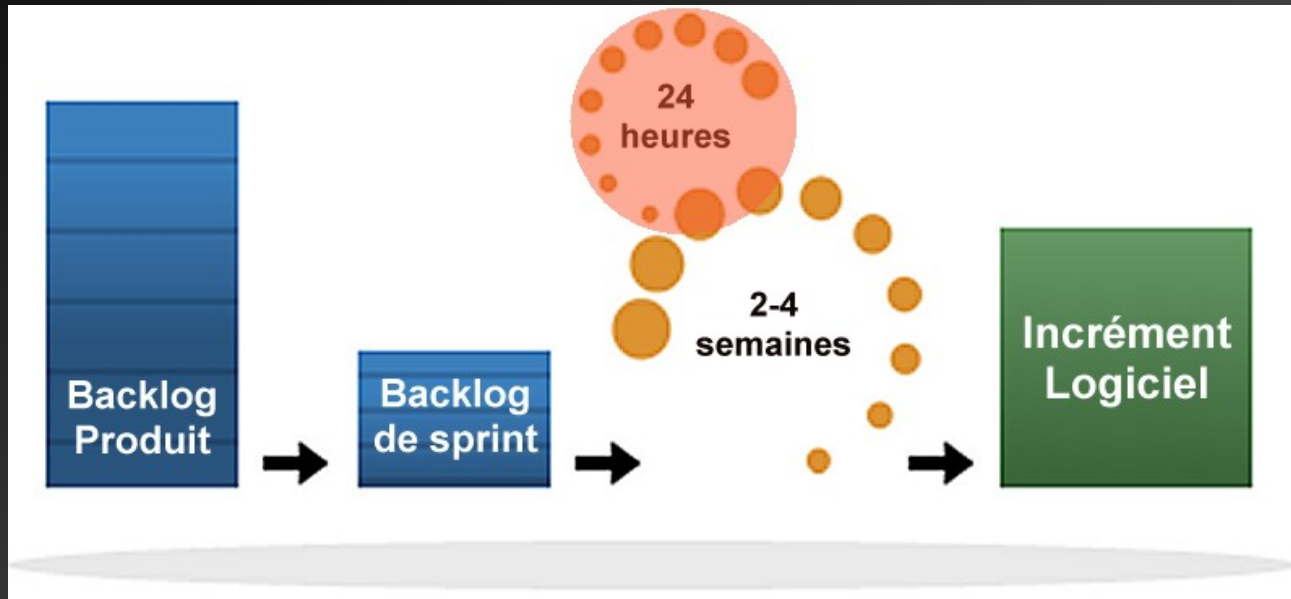


Source : [www.scrumalliance.org](http://www.scrumalliance.org)

## 3. Sprint

- Développement des fonctionnalités du backlog de sprint
- Aucune modification du backlog de sprint possible

# Scrum – Organisation 4/5

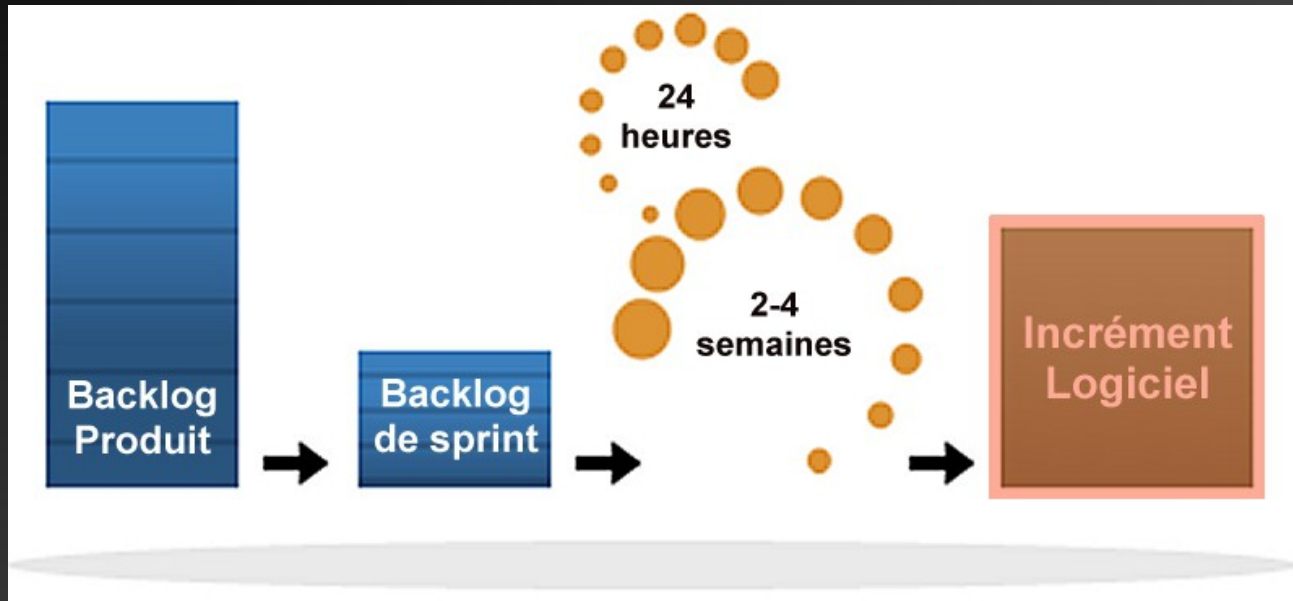


Source : [www.scrumalliance.org](http://www.scrumalliance.org)

## 4. Mêlée quotidienne

- Point de contrôle quotidien de l'équipe
- Interventions régulées – 2 min. par personne

# Scrum – Organisation 5/5

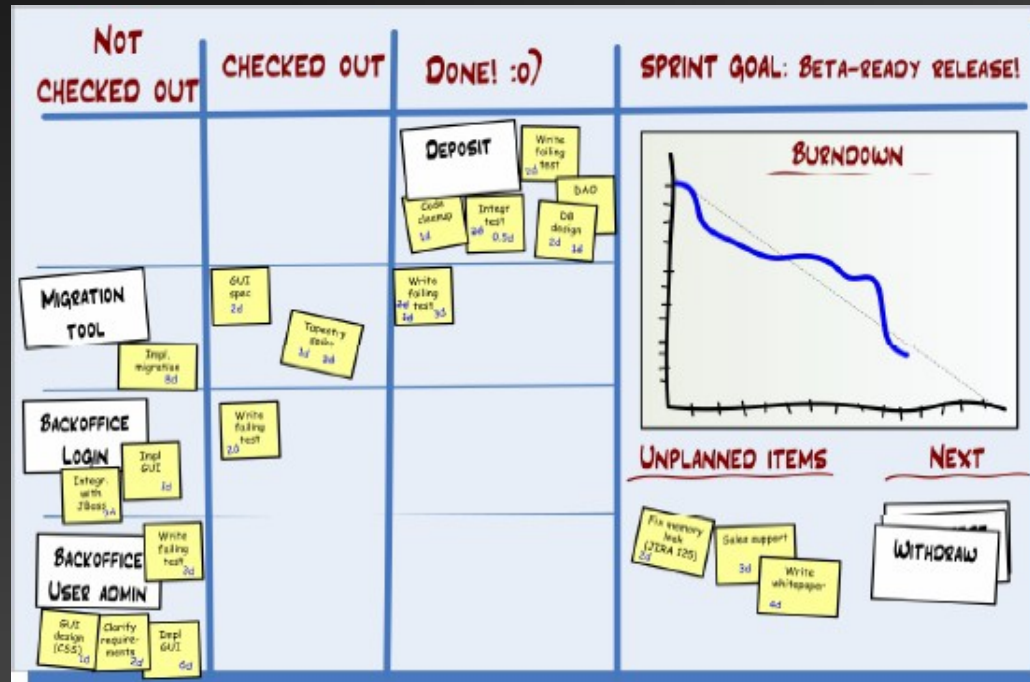


Source : [www.scrumalliance.org](http://www.scrumalliance.org)

**5. Incrément logiciel** : livré au product owner à la fin du sprint.

# Scrum – Indicateurs de projet 1/2

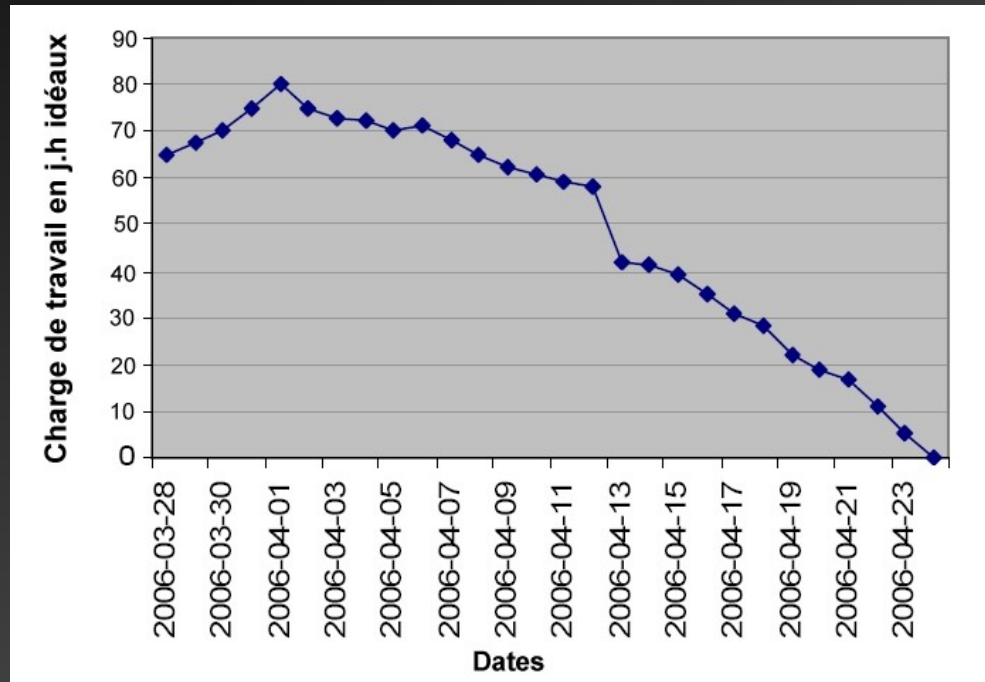
- Le tableau des tâches



Source : « Scrum and XP from the trenches » de H. Kniberg, 2007

# Scrum – Indicateurs de projet 2/2

## ■ Le burndown chart



Source : « Summary of Scrum », Signifikant Svenska A.B., 2007

# Scrum – Ingénierie logicielle

---

- Scrum est une méthode de gestion de projet
- Doit être complétée par des techniques d'ingénierie logicielle
- Complémentaire avec eXtreme Programming :
  - Test Driven Development
  - Intégration continue

# Scrum – Equipes plus grandes

---

- Principes :
  - Commencer par une équipe Scrum standard
  - Création de plusieurs équipes – essaimage
- Adaptation de la méthode :
  - Scrum des scrums
  - Rôle de team lead
- Problèmes à traiter :
  - Dispersion géographique
  - Développement off-shore



# Les outils

---

## ■ Outils traditionnels

- Tableau blanc et post-its
- Excel – Backlog produit et backlog de sprint

## ■ Outils dédiés

- Outils commerciaux / Open source
- Gèrent une charge de travail
- Absence de PERT / Gantt
- Intégration avec : IDE, contrôle de sources, gestion des tests, bug tracking, intégration continue.

## ■ Autres outils

- Connexion large bande
- Wiki, webcams, messagerie instantanée...

# Perspectives

---

- Pas d'évolution, peu de critiques
- Défauts à palier
  - Absence de dépendance entre les tâches
  - Polyvalence des programmeurs
  - Productivité équivalente supposée
  - ⇒ Grande **maturité** nécessaire
- Contrats à adapter
- Stratégie d'introduction de Scrum en entreprise

# Conclusion

---

- Méthode de gestion de projet – développement logiciel
- A compléter avec des techniques d'ingénierie logicielle
- Rien de totalement nouveau
- Méthode à la mode. Conditions propices nécessaires
- Expérimentations prometteuses
- Principal bénéfice : des équipes motivées



Questions ?

---