L’agilité : de la promesse à la réalité !

Séminaires animés par Objet Direct
Le 17 juin 2010 à Lyon
Le 22 juin 2010 à Paris

OBJET DIRECT
Architectures Web & Objet • Groupe Viseo

Objet Direct, spécialiste objet & web

www.objetdirect.com
http://blog.objetdirect.com

OBJET DIRECT
Architectures web & objet • Groupe Viseo

Notre métier

Comment aligner le SI sur les processus métiers de l'entreprise ?

Comment industrialiser nos développements applicatifs ?

Quels outils, quels frameworks choisir ?

Quelle démarche projet, quelle méthode pour améliorer nos processus projets ?

Comment former efficacement nos collaborateurs ?

Comment moderniser notre patrimoine applicatif ?

Depuis 12 ans, Objet Direct répond concrètement à ces questions avec des prestations d'expertise dans le domaine du génie logiciel objet.

Objet Direct, spécialiste Web et objet une offre IT complète

Offres de conseil IT

- Audit et schéma directeur du SI
- Architecture d'entreprise, urbanisation du SI
- Audit d'applications
- Conseil méthodologique
- Conseil au choix d'architecture et d'outils
- Conseil en recrutement et gestion des compétences

Réalisation de projets

- Assistance à maîtrise d'ouvrage
- Analyse, conception et modélisation UML
- Assistance technique spécialisée
- Réalisation de projets en maîtrise d'œuvre
- Centres de services agiles
- Tierce maintenance applicative

Formation, coaching

- Une offre de cours complète autour des disciplines du développement orienté objet
- Formations animées en inter et intra-entreprise
- Ingénierie de formation : production de cours adaptés au contexte client
- Coaching post-formation sur les projets

Objet Direct, spécialiste Web et objet
Centres d'expertise et compétences

Agilité

Méthodes agiles :

Scrum

UP - Unified Process

XP - eXtreme Programming

Lean Software Development

Modélisation

UML

Référentiels, BPM

AMOA

Conception

Design Patterns

Modeleurs et AGLs

Architecture

n-tier - Web – RDA

Cloud Computing
SOA , Web Services
Frameworks
Mapping O/R, IoC,
NoSQL, Clustering
SSO

Open Source

Outillage

Intégration continue
Test automatisé
TDD, Cockpit project
Environnements
Collaboratifs
Maven, Ant
Hudson

WEB 2.0

RIA : Flex, Silverlight
GWT , JSF,
Ajax, Wicket, Struts
Dojo, jQuery,
Java / JEE
PHP5, Ruby-on-Rails

.NET

Visual Studio 2010
Team Foundation Server
Frameworks ASP.Net
WCF – WPF – WWF
Langages C#,
VB .NET, C++

Mobile

iPhone
Smartphone /Android
Windows Mobile
Objective-C / xCode
Cocoa Touch
Interface Builder

Model Driven

Model Driven Engineering
Prototypage, D.OM
Chaines de transformation de modèles
Génération de code
Domain Driven Design
Modernisation SI

Objet Direct, spécialiste Web et objet
Quelques références

BANQUE-ASSURANCES : BANQUE PRIVEE 1818, CACEIS, GIE AMT / CEDICAM / MONECAM (Groupe CREDIT AGRICOLE), LA BANQUE POSTALE, NATIXIS…

ENERGIE/ CHIMIE/ INDUSTRIE/ PHARMA : AGENCE DE L'EAU, ALCAN, ALLIANCE HEALTHCARE, BIOMERIEUX, EDF, CEGELEC, INSTITUT FRANÇAIS DU PETROLE, LABORATOIRES BOIRON, PSA, RENAULT, STMICROELECTRONICS…

SECTEUR PUBLIC : AMF (Autorité des Marchés Financiers), ARMEE DE TERRE, BRGM (Bureau de Recherches Géologiques et Minières), CEA, CHU Grenoble, CONSEIL D'ETAT, ESRF (Synchrotron de Grenoble), RECTORAT DE GRENOBLE…

SERVICES : AKERYS, CANAL+, DRIMKI, GECINA, TELEM (groupe ONET)…

TELCO/ ELECTRONIQUE/ HI-TECH : BOUYGUES TELECOM, DILICOM, ESERV GLOBAL, FRANCE TELECOM, HP , KELKOO, KOMPASS, LA ROUE VERTE, SFR, STARZIK, VOLUBILL, XEROX, YAHOO FRANCE, YOOGET…

TRANSPORT/ LOGISTIQUE : CORYS, GEODIS, GROENEVELD, GTIE, VOYAGES-SNCF.COM…

Groupe VISEO
Spécialiste de votre système d'information

Conseil architecture du SI

Applications métiers spécifiques

Web et Objet

Optimisation et intégration des processus

ERP

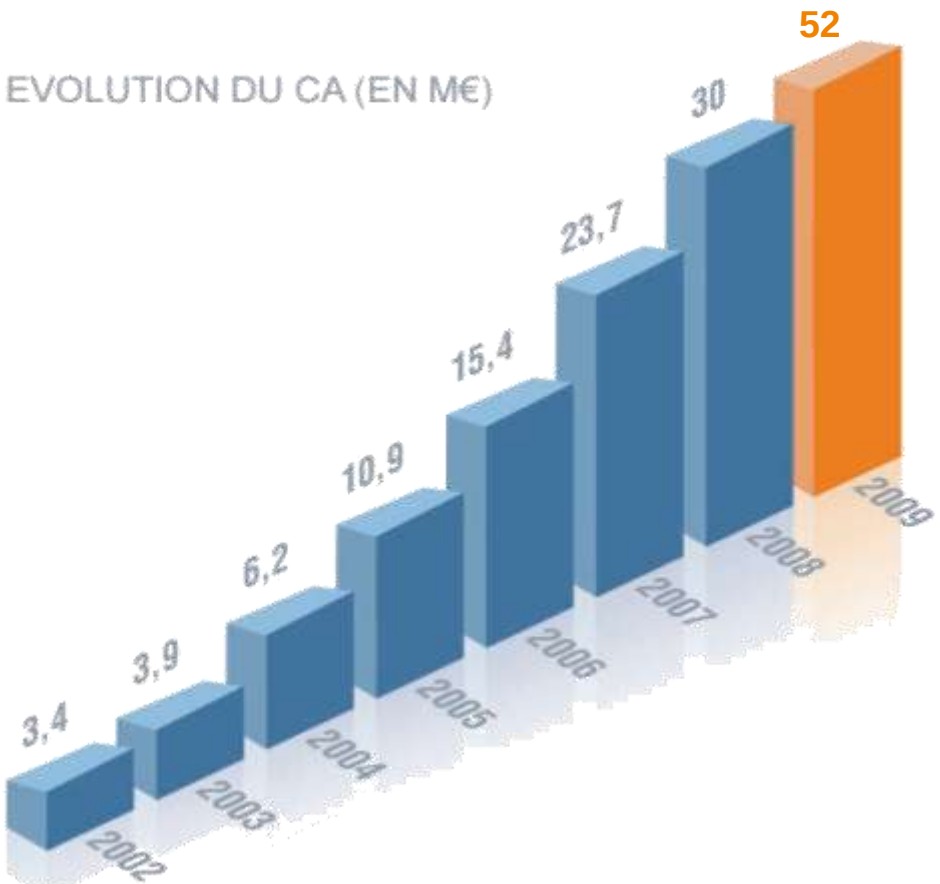
Pilotage de l'entreprise

Optimisation performance Financière

BI

CONSEIL AUDIT / DIAGNOSTIC → **VENTES DE LICENCES** → **DEVELOPPEMENT INTEGRATION** → **TIERCE MAINTENANCE APPLICATIVE**

VISEO : les chiffres clés



VISEO
Une présence sur chaque continent

Un maillage régional pour favoriser la proximité avec nos clients

Les implantations de VISEO permettent d'accompagner les clients dans leurs **projets internationaux** et d'adresser des clients locaux.

L'offre de **prestations offshore**, au Maroc, propose des développements spécifiques ABAP.

Contacts
www.objetdirect.com – http://blog.objetdirect.com

Objet Direct Paris - siège
10 rue Lincoln - 75008 Paris -Tél. 01 76 69 16 09

Olivier Lefaucheux - Directeur d'agence
olefaucheux@objetdirect.com
Laure Van de Walle - IA - lvandewalle@objetdirect.com
Stéphane Jotic - IA - sjotic@objetdirect.com

Objet Direct Lyon
1, cours Albert Thomas - 69003 Lyon -Tél. 04 72 33 78 30

Emmanuel Pires - Directeur d'agence
epires@objetdirect.com
Hervé Jacob - Resp cial - hjacob@objetdirect.com
Jocelyn Dupré - IA - jdupre@objetdirect.com

Objet Direct Grenoble
34, avenue Félix Viallet - 38000 Grenoble - Tél. 04 76 53 72 72
Thibault Parmentier - Directeur d'agence
tparmentier@objetdirect.com
Anne Hardyau - IA - ahardyau@objetdirect.com
Marjorie Melenotte - IA - mmelenotte@objetdirect.com

Formation
Centres de Lyon et Grenoble -Tél.: 04 72 33 78 30

Laurence Perret - Responsable
lperret@objetdirect.com

Agences de Toulouse & Bordeaux
9, rue Paulin Talabot - 31100 Toulouse - Tél. 05 34 60 38 30
Richard Pont - Directeur d'agence
rpont@homsys.com
Arnaud Fouquet - IA - afouquet@homsys.com

Agence de Marseille
42, rue des Mousses - 13008 Marseille - Tél. 04 91 16 01 26

Richard Pont - Directeur d'agence
rpont@homsys.com
Adriana Caldas - IA - acaldas@homsys.com

L’agilité : de la promesse à la réalité !

Le 17 juin 2010 à Lyon
Le 22 juin 2010 à Paris
Séminaires animés par Henri Darmet, directeur technique

OBJET DIRECT
Architectures web & objet • Groupe Viseo

Agenda

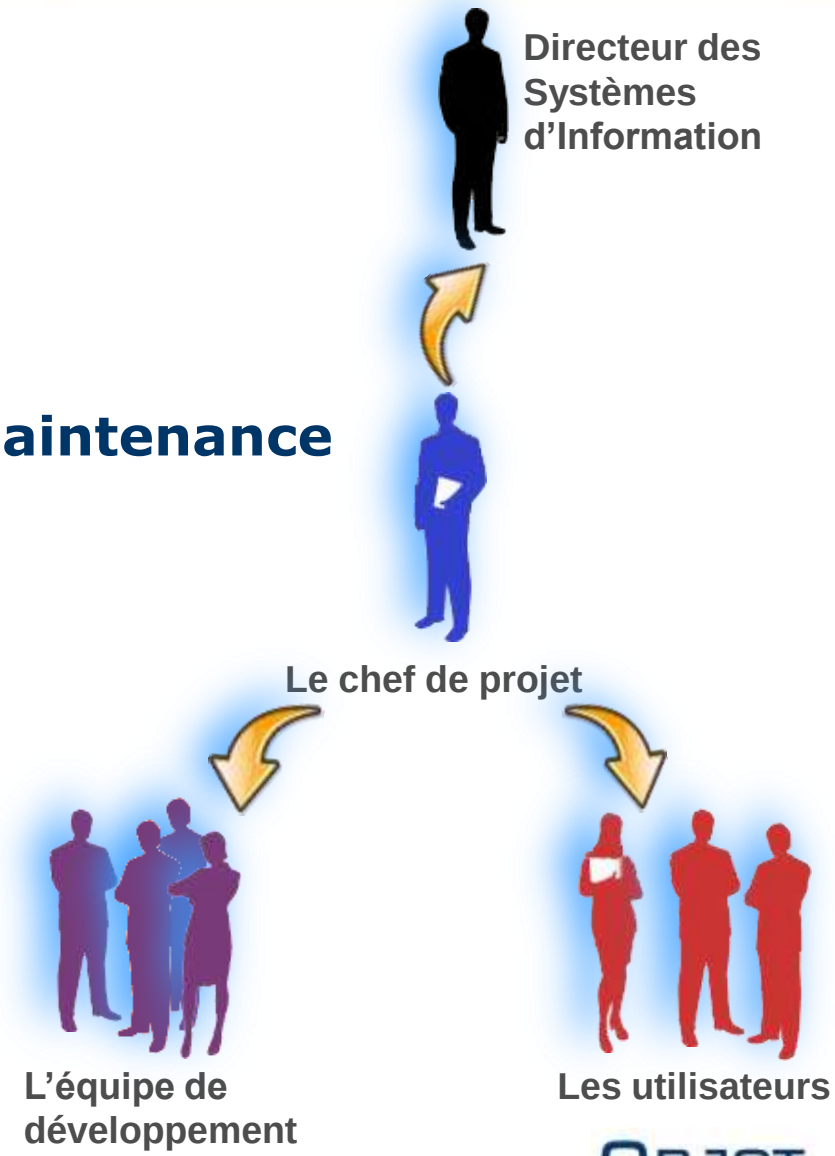
- Pourquoi faire de l'agilité ?
- Les piliers de l'agilité
- Unified Process
- Extreme Programming
- Scrum
- Les pièges et les bonnes pratiques
- Conclusion

Agenda

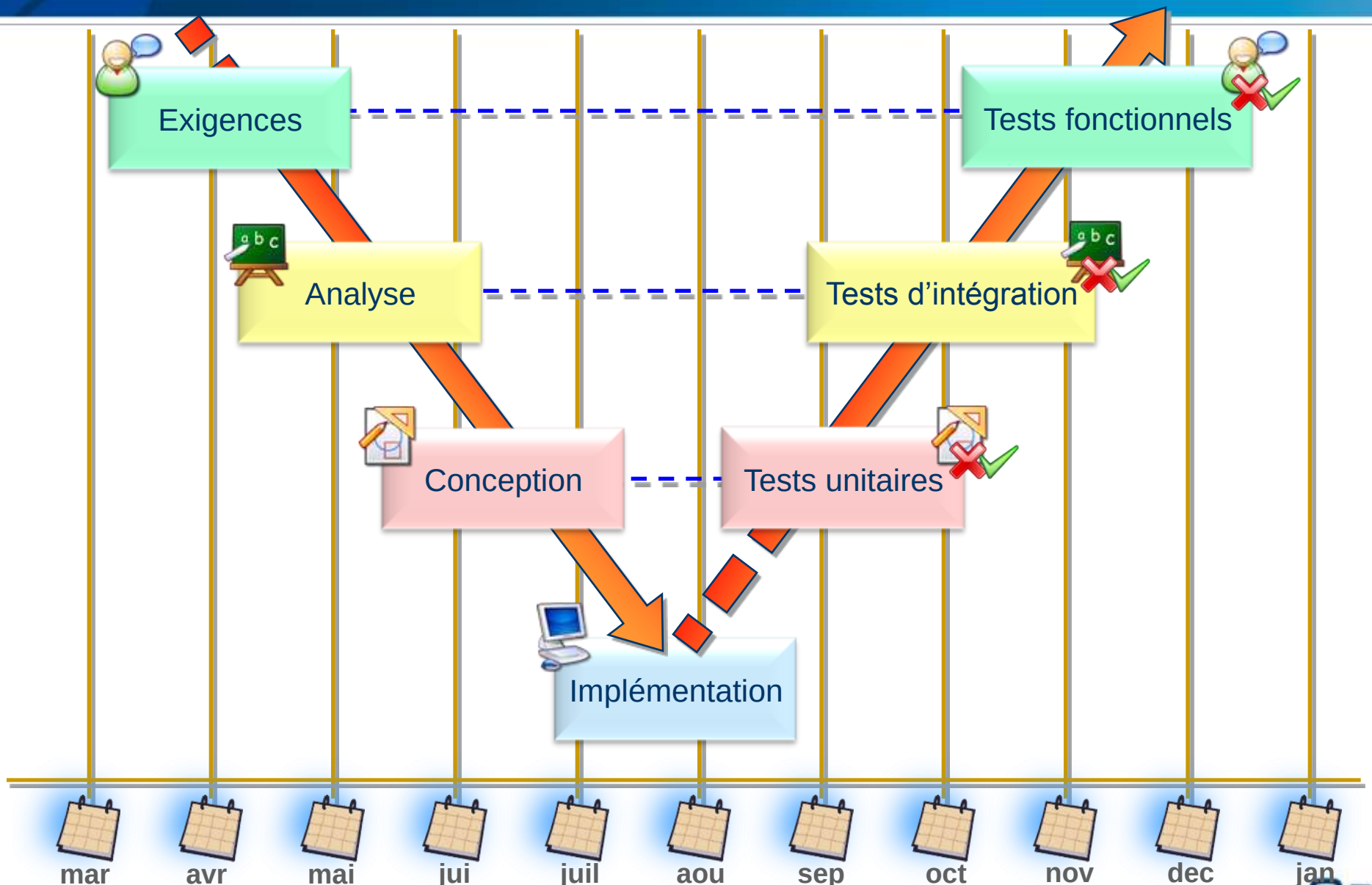
- **Pourquoi faire de l'agilité ?**
- Les piliers de l'agilité
- Unified Process
- Extreme Programming
- Scrum
- Les pièges et les bonnes pratiques
- Conclusion

Les défis du projet

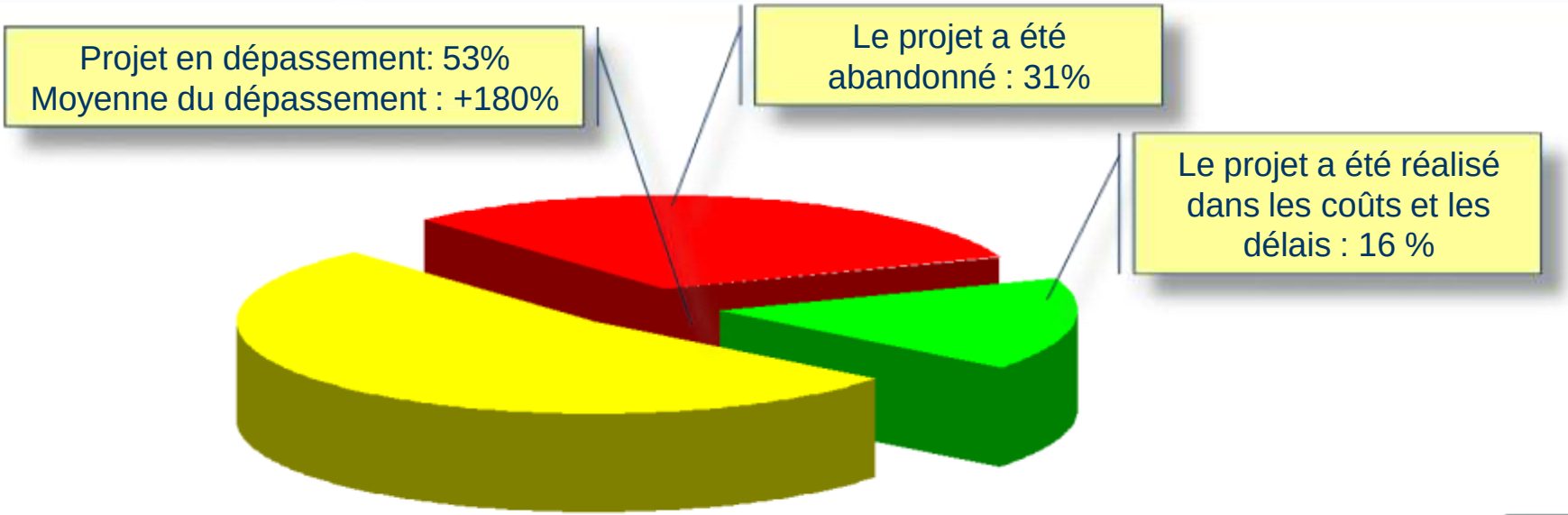
- **Répondre au besoin des utilisateurs**
  - Livrer un logiciel utile et utilisable
    - Utile : il répond au besoin
    - Utilisable : il propose une ergonomie adaptée
  - Livrer à temps
- **Limiter les coûts de production et de maintenance**
  - Utiliser efficacement les ressources disponibles
    - Equipe, outils
  - Ne produire que ce qui est utile
    - Et uniquement cela
  - Ne pas sacrifier la qualité
    - Afin de garantir la maintenabilité
- **Donner de la visibilité à son projet**
  - Pour le management
  - Pour l'équipe
  - Pour l'utilisateur final



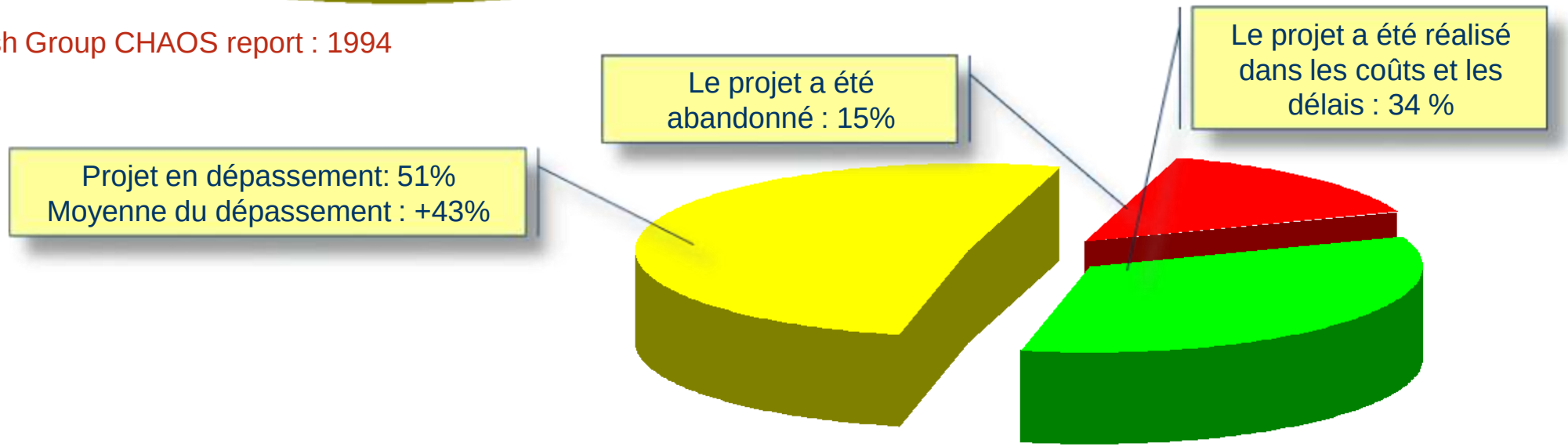
Le cycle en cascade et en V



Pour quel résultat ?



Standish Group CHAOS report : 1994

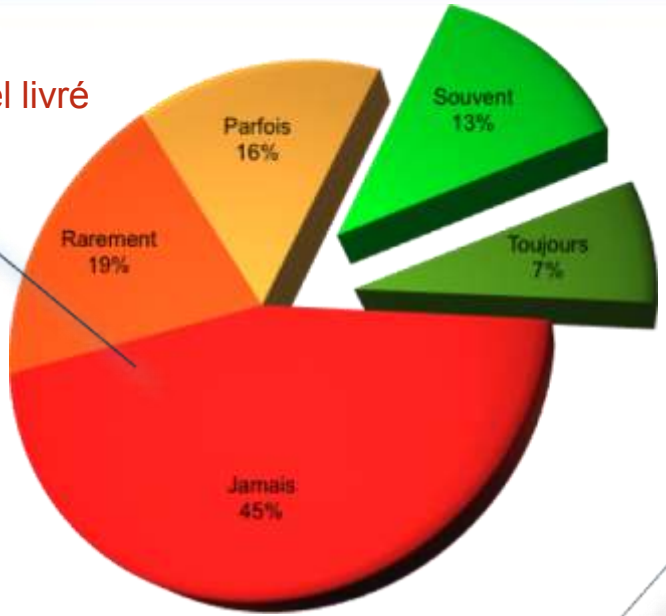


Standish Group CHAOS report : 2003

Pourquoi ?

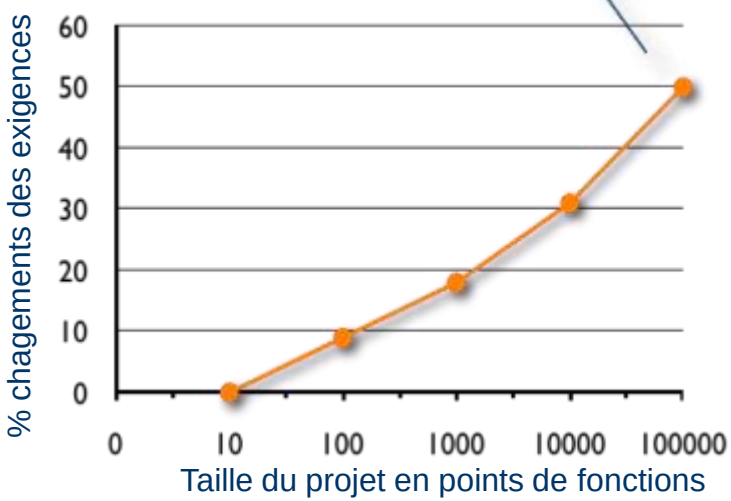
Fonctions utilisées d'un logiciel livré

Près de la moitié des fonctions ne sont jamais utilisées !



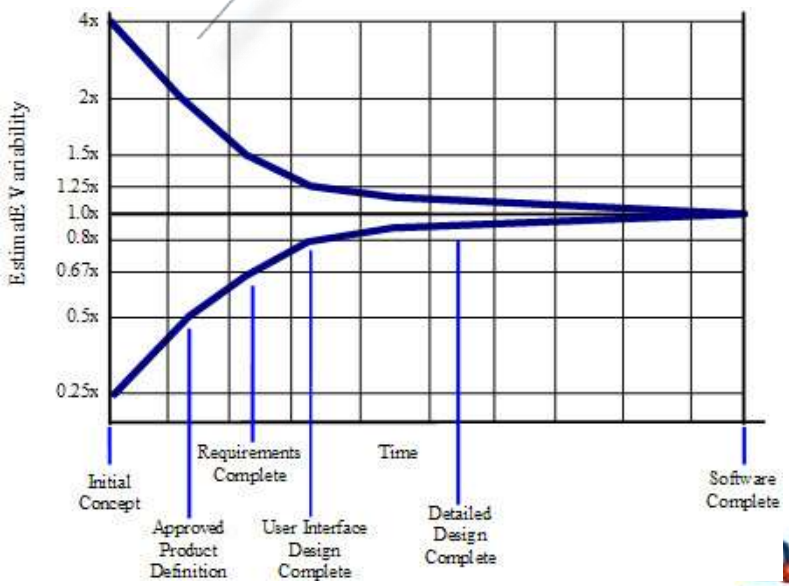
Le besoin d'un logiciel d'entreprise évolue de 25% à 50%

Évolution du besoin



Il faut avoir terminé la conception pour pouvoir construire une planification fiable !

Cône d'incertitude



Il est donc nécessaire :

- **D'établir un dialogue de qualité avec l'utilisateur**
  - Comprendre ce qu'il dit
  - Parler un langage qu'il comprend
  - L'impliquer dans la réussite du projet
- **De s'organiser en s'appuyant sur des estimations fiables**
  - Un plan précis mais faux, est plus qu'inutile : il est dangereux !
  - Il faut estimer la qualité de l'information dont on dispose
    - Une projection sur 3 semaines est très fiable. Cette fiabilité décroit ensuite
    - Compter les fonctions testées et intégrées a plus de valeur qu'estimer un reste à faire
- **D'optimiser le travail de l'équipe de développement**
  - Faire ce qui est nécessaire et suffisant pour l'utilisateur
  - Minimiser les productions intermédiaires
  - Favoriser le partage des savoirs et des compétences

Agenda

- Pourquoi faire de l'agilité ?
- Les piliers de l'agilité
- Unified Process
- Extreme Programming
- Scrum
- Les pièges et les bonnes pratiques
- Conclusion

Le manifeste des méthodes agiles

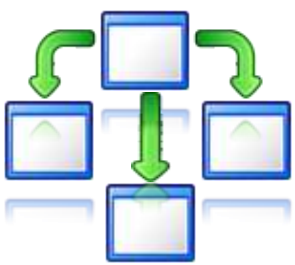
- **Août 2001, publication du manifeste des méthodes agiles**
  - Par 17 personnalités éminentes du développement
    - Ward Cunningham (Wiki)
    - Kent Beck (eXtreme Programming), Ken Schwaber, Jeff Sutherland (SCRUM)
    - Alistair Cockburn, Martin Fowler

- **Les valeurs de l'agilité**
  - Parier sur les hommes et la communication
    - plutôt que le processus ou l'outillage
  - Ecrire du logiciel qui fonctionne
    - plutôt que de rédiger de la documentation
  - Collaborer avec l'utilisateur
    - plutôt que d'établir une relation contractuelle
  - Réagir et s'adapter aux changements
    - plutôt que de suivre un plan pré-établi

Parler aux utilisateurs

- **Document de spécification fonctionnelle**
  - Trop formel => informaticiens
  - Long, verbeux, obscur, abstrait

- **Maquette**
  - Une application complètement factice
  - Des écrans articulés autour d'une navigation statique
  - Très utile pour valider l'aspect statique (écran)
  - Moins bon pour l'aspect dynamique

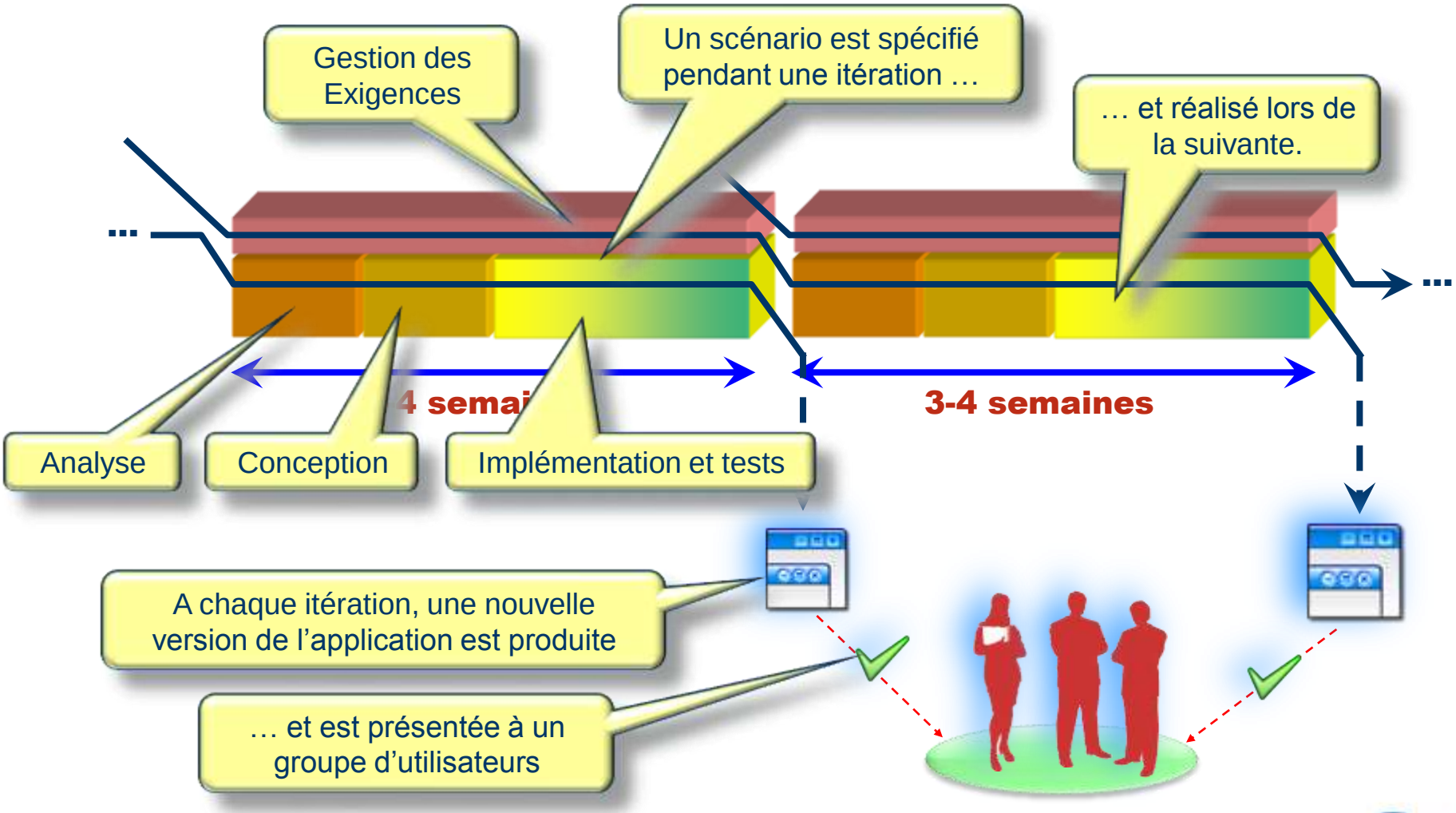


- **Story board**
  - L'animation d'un scénario
  - Une suite de slides
  - Très bon pour les aspects dynamiques et statiques

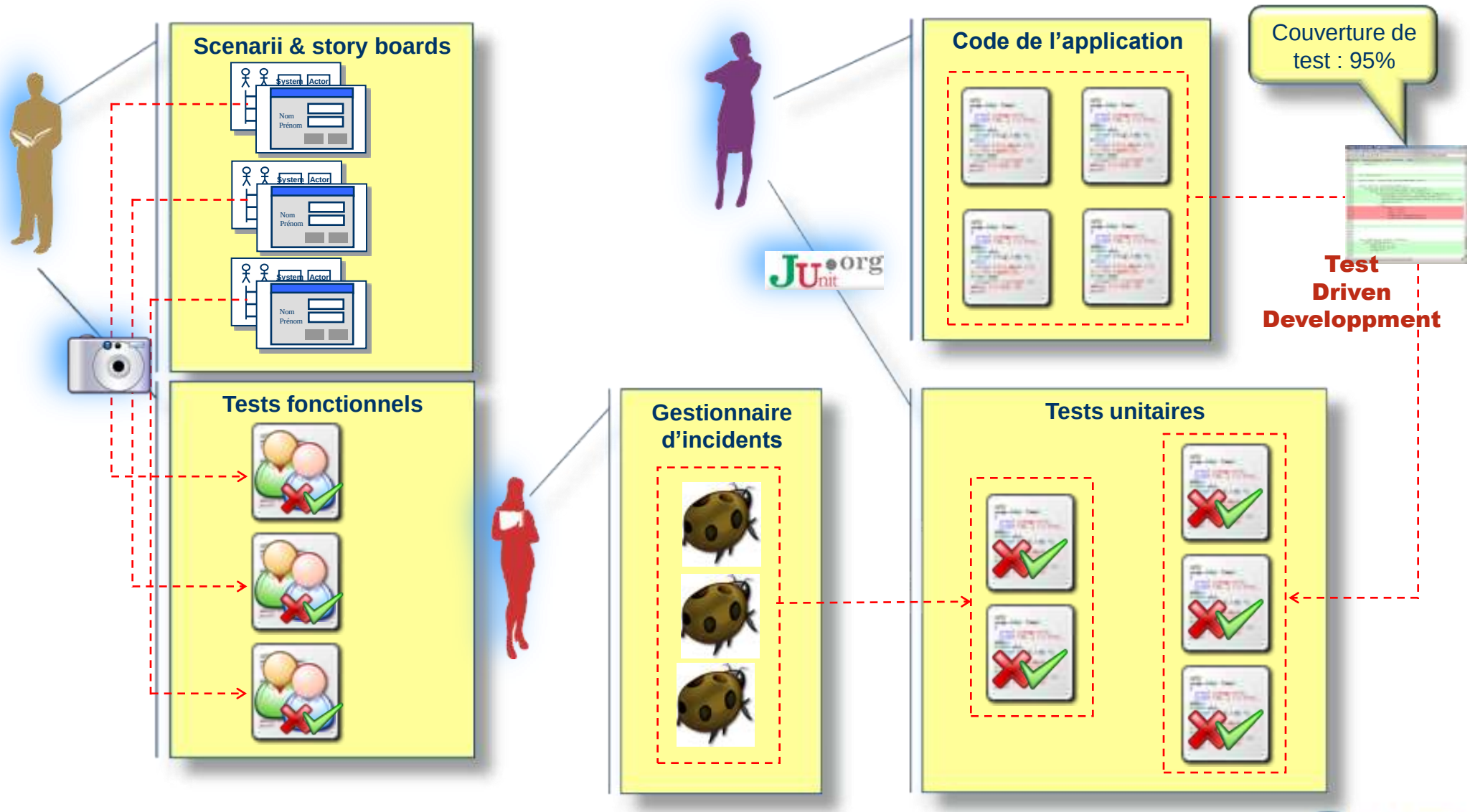
- **Prototype fonctionnel**
  - Application jouet utilisable
  - Générée à partir d'une spécification (UML)
  - Excellent pour valider les règles métier
  - Peut aussi servir pour qualifier l'ergonomie

L'itération

- **Mini-projet de quelques semaines (1 à 4)**



Etablir une stratégie de test

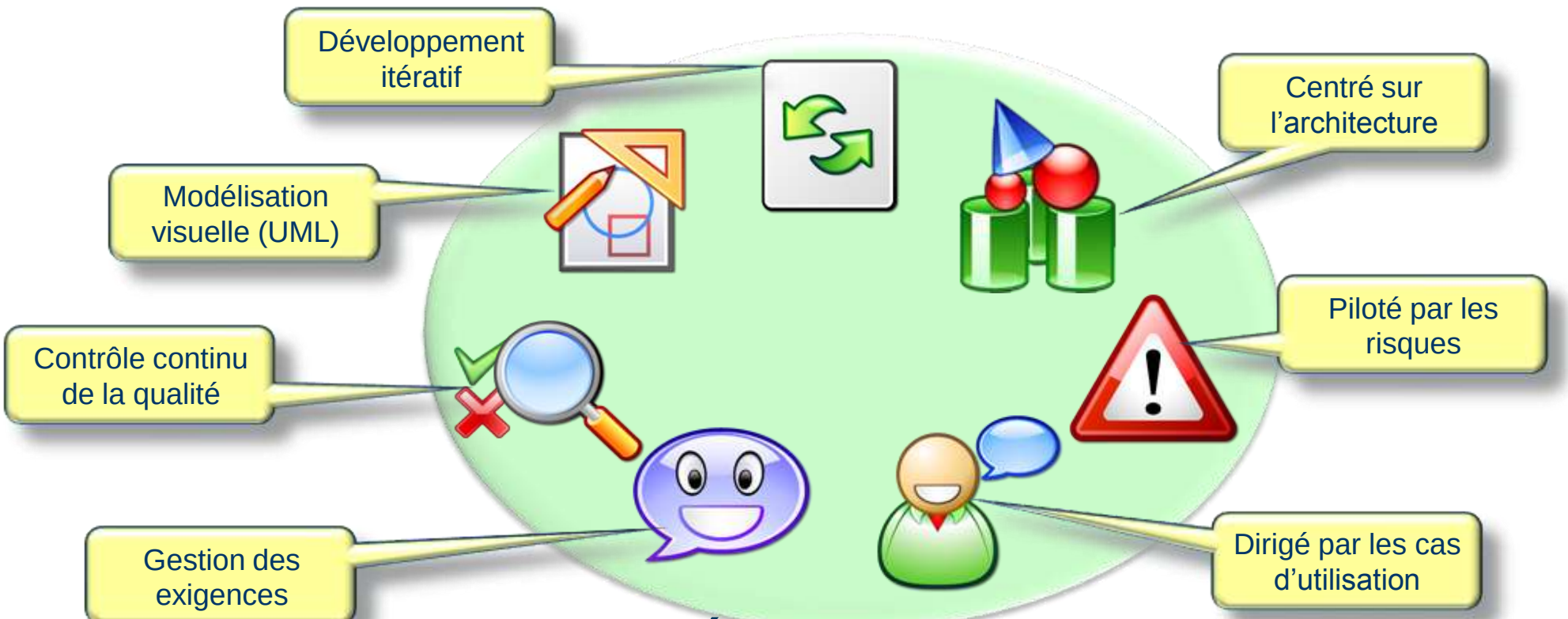


Agenda

- Pourquoi faire de l'agilité ?
- Les piliers de l'agilité
- Unified Process
- Extreme Programming
- SCRUM
- Les pièges et les bonnes pratiques
- Conclusion

Le Processus Unifié

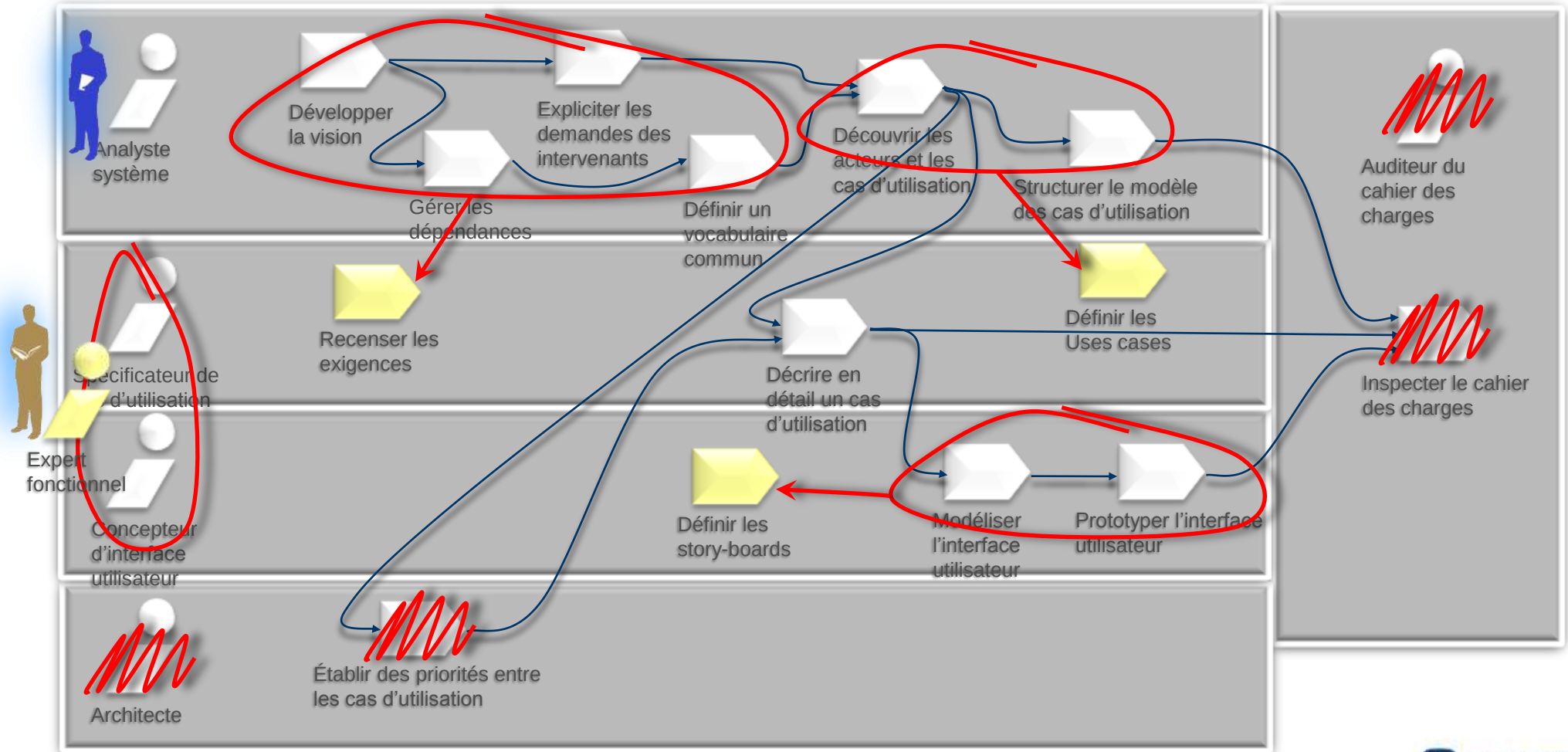
- **Guide méthodologique pour la fabrication de logiciels**
- **Élaboré par des experts reconnus sur la base de l'expérience à la fin des années 1990**
- **Ensemble cohérent de bonnes pratiques dont :**



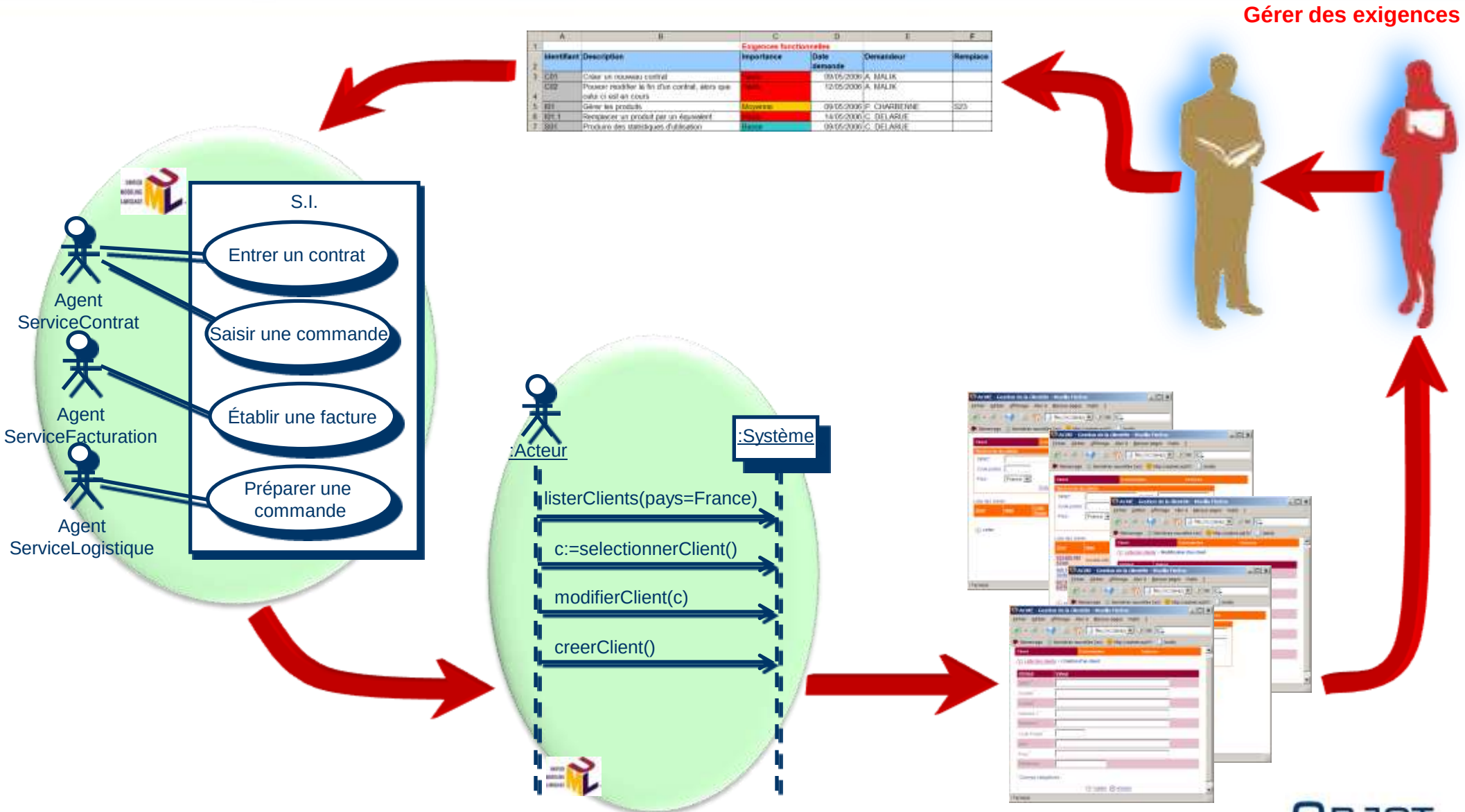
- **Efficace lorsqu'il est associé aux technologies objet**

Adapter le Processus Unifié

- UP peut être utilisé pour des projets de toutes tailles, de tous types
- Il faut l'adapter au contexte du projet
- Exemple : les activités liées aux exigences (spécs fonctionnelles)

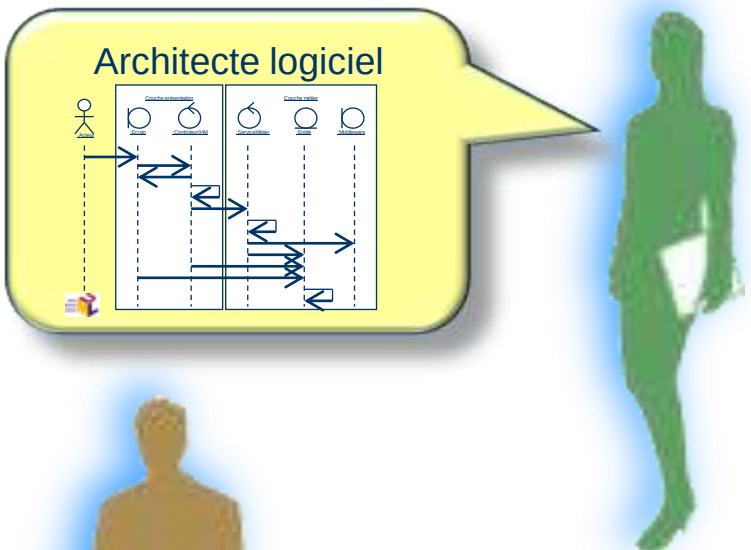


Processus de prise en compte des besoins

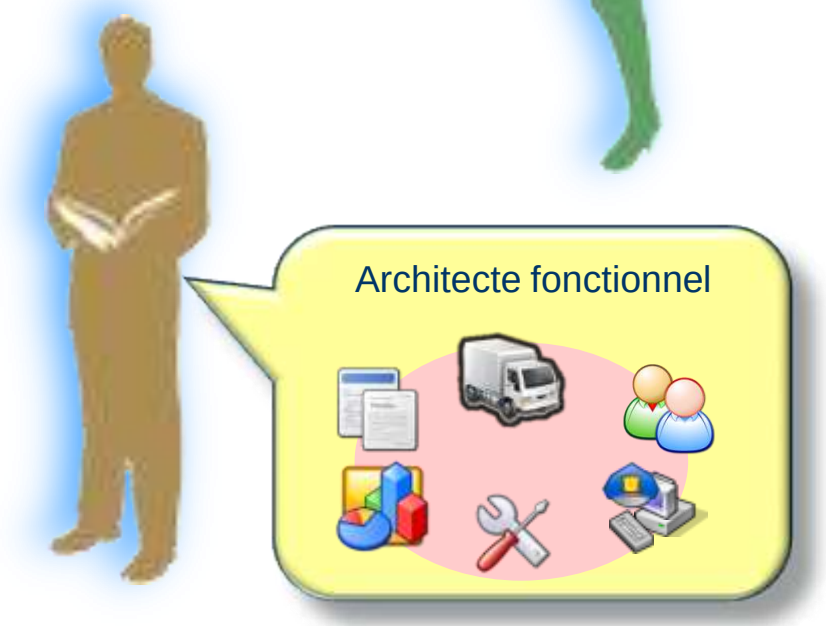


S’appuyer sur une architecture

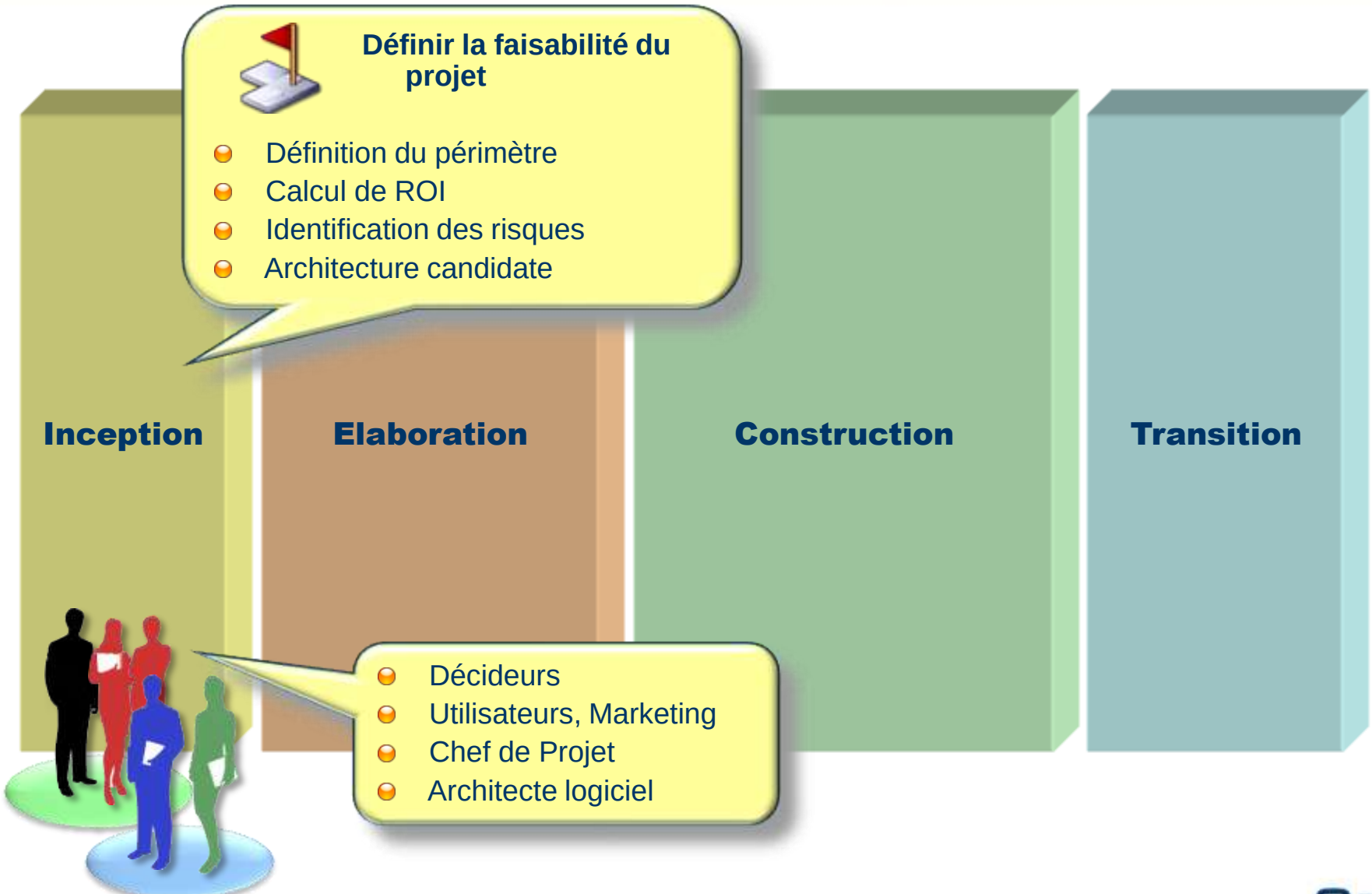
- **L’architecture en informatique, c’est :**
  - L’architecture fonctionnelle
    - Définition des processus métier
    - Découpage en modules fonctionnels
  - L’architecture logicielle
    - Découpage en couches
    - Typologie des composants
    - Choix des technologies utilisées



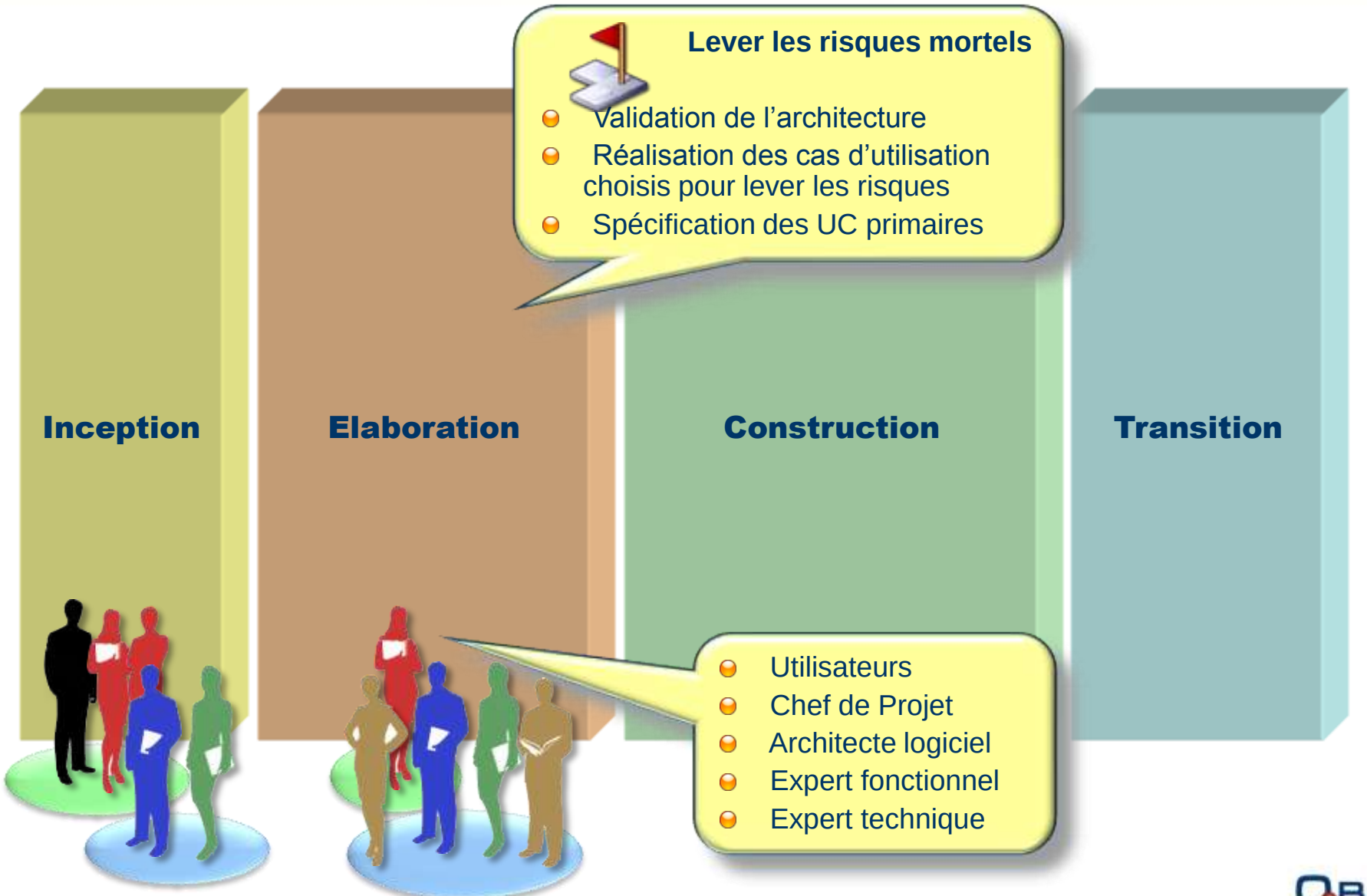
- **Les vertus de l’architecture**
  - C’est une preuve de faisabilité
  - C’est un exemple à reproduire
  - Elle structure et organise
  - Elle permet d’affronter rapidement le risque



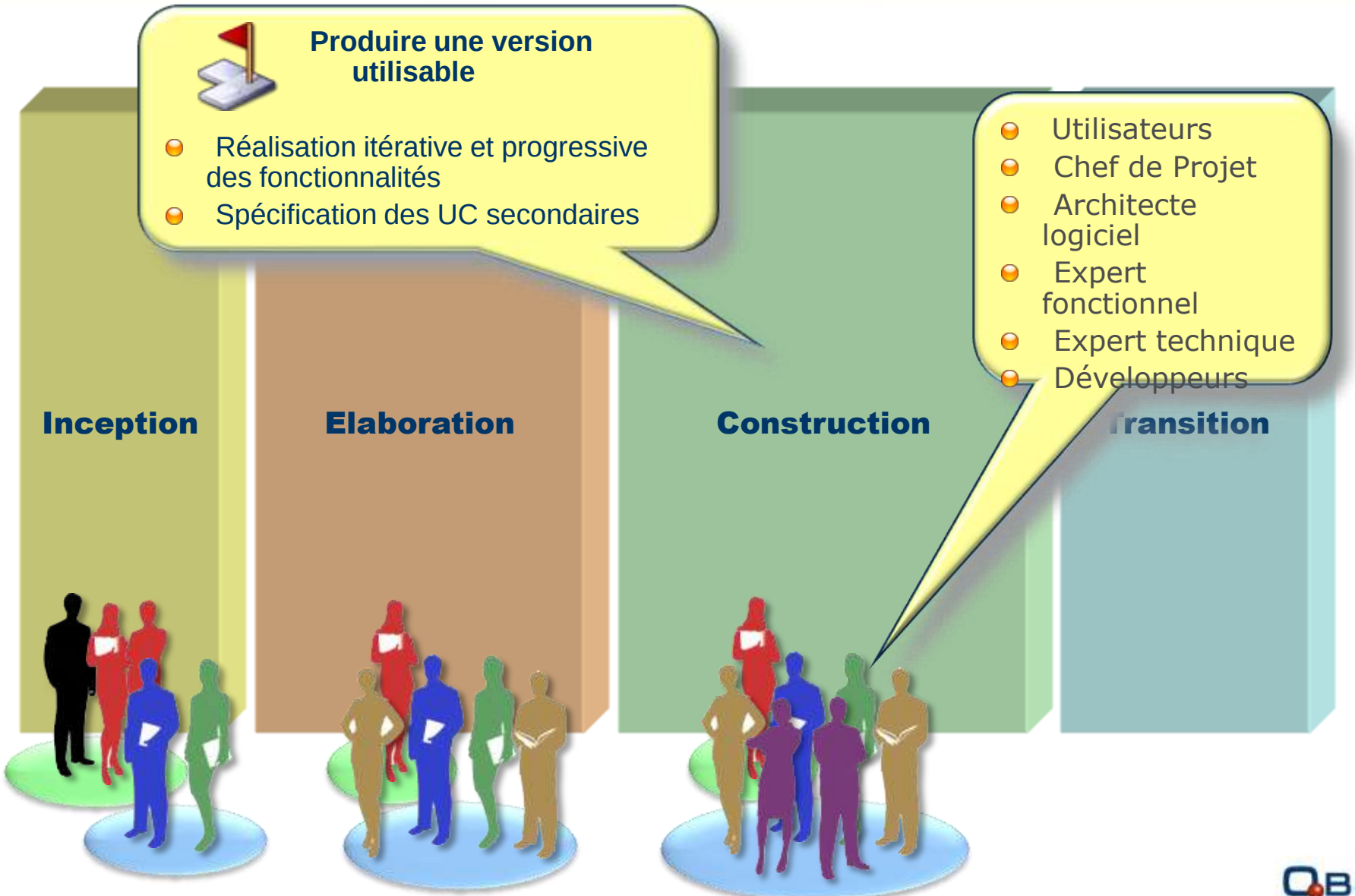
Principe d'organisation : les phases



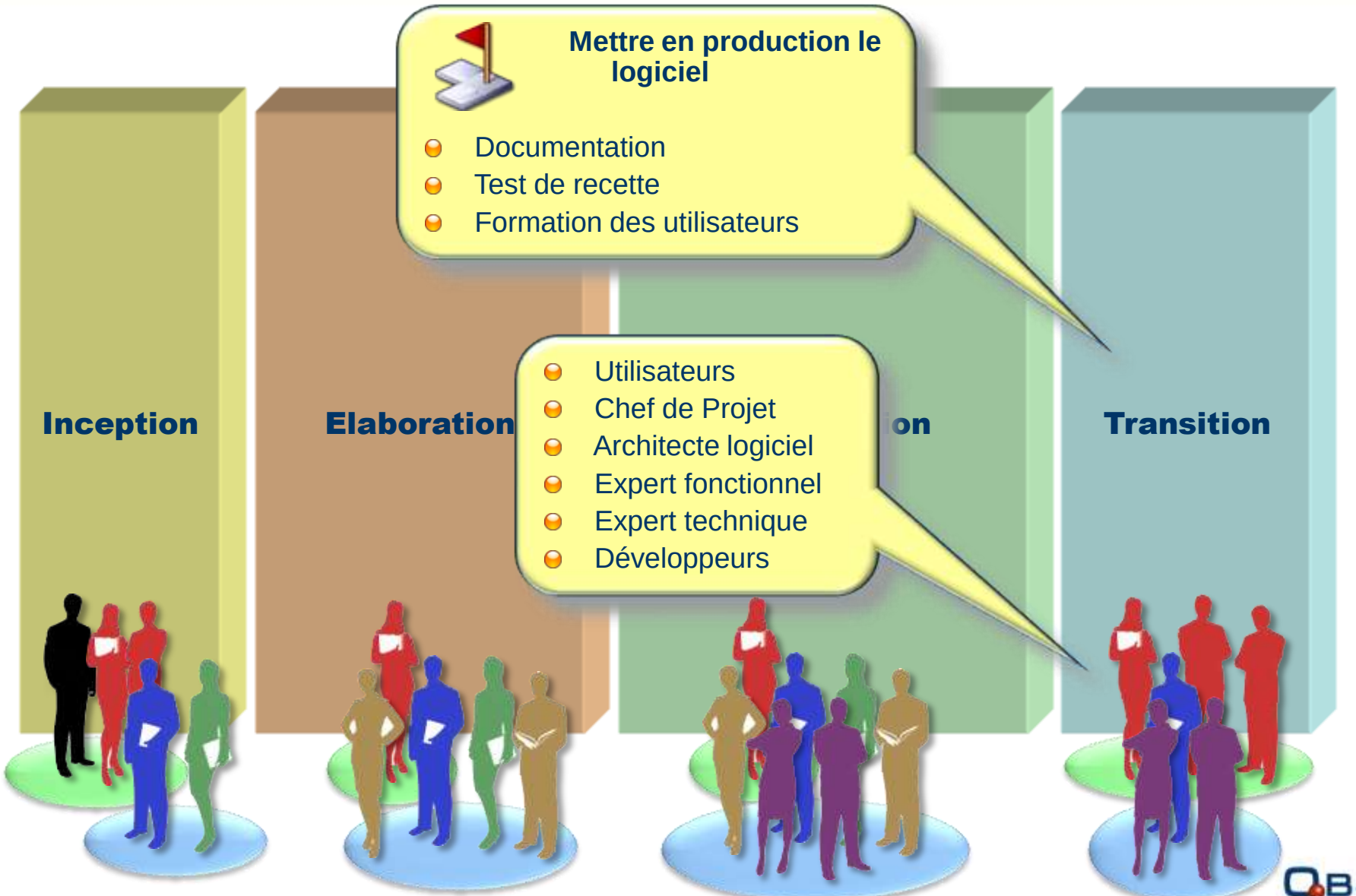
Principe d'organisation : les phases



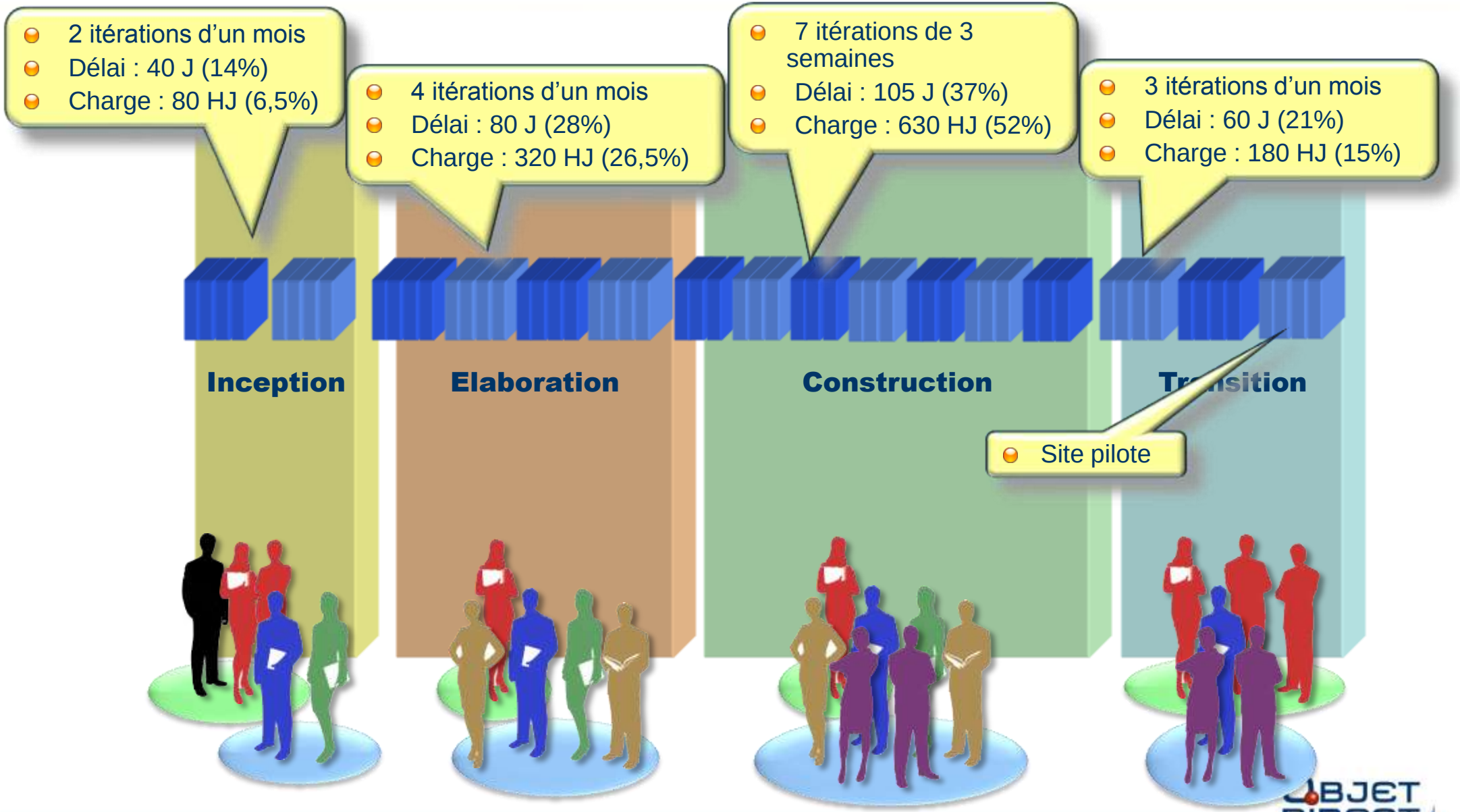
Principe d'organisation : les phases



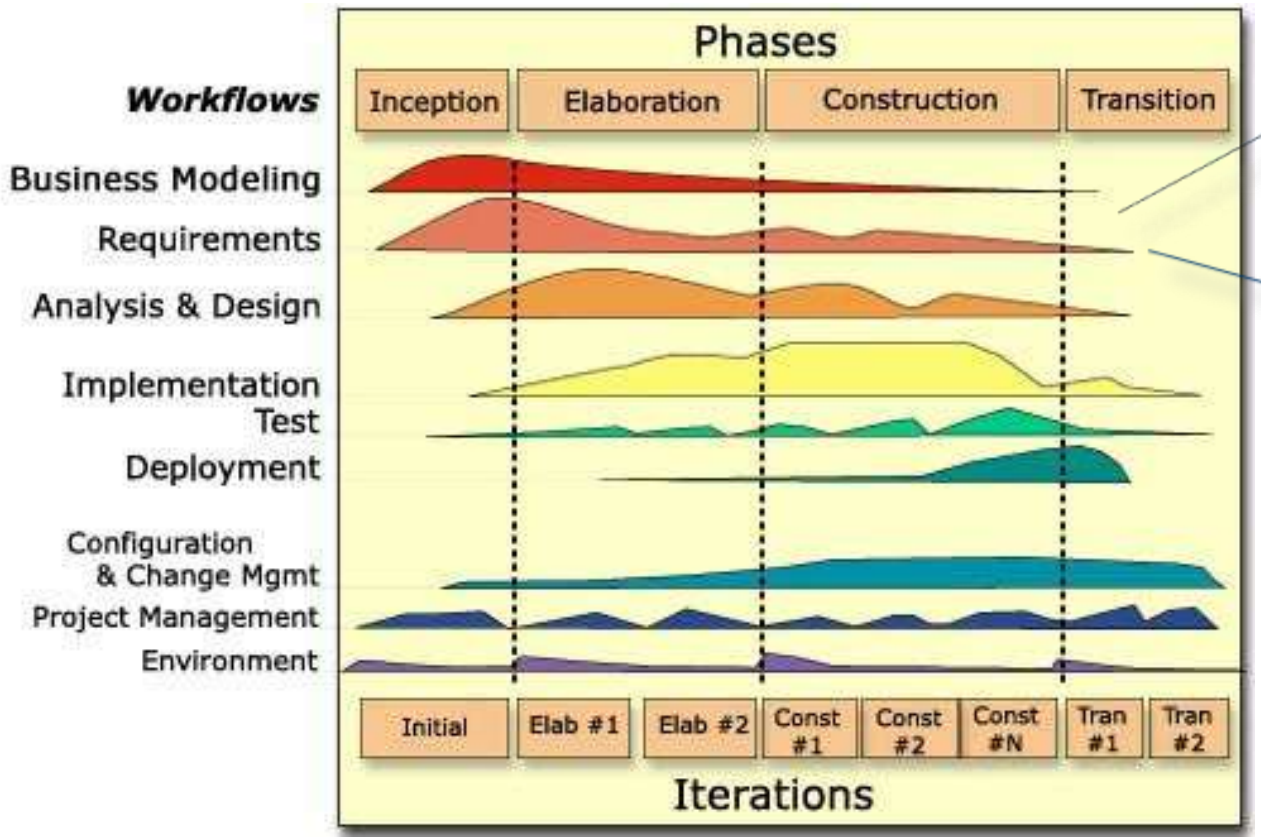
Principe d'organisation : les phases

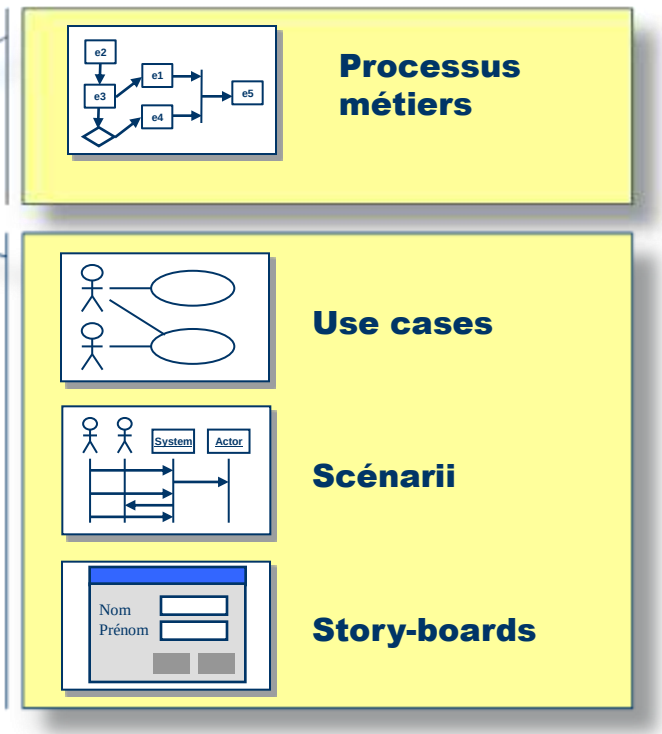


Instanciation d'UP pour un projet de 1000 jours

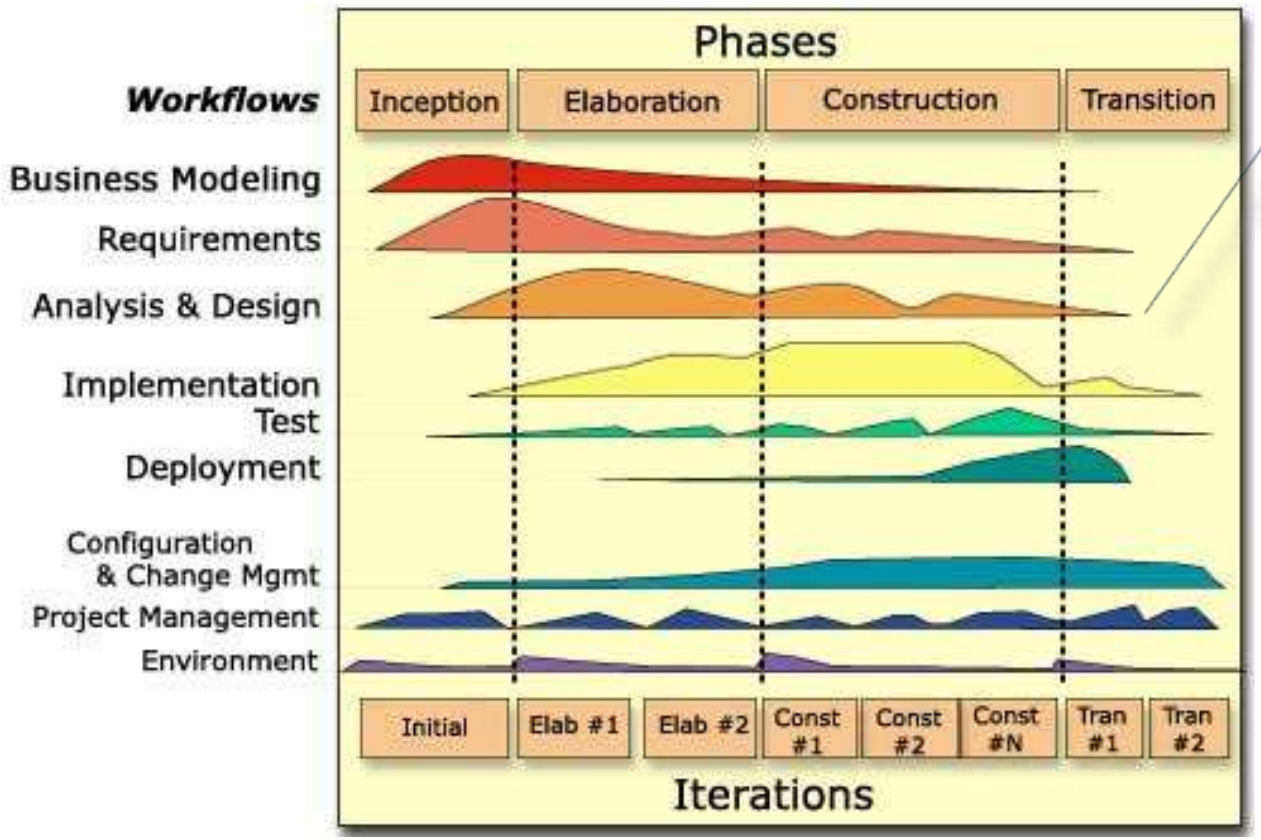


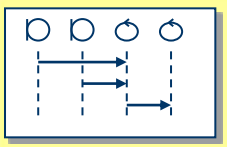
Principes d'organisation : les disciplines



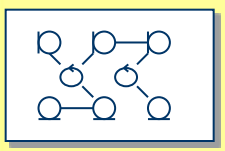


Principes d'organisation : les disciplines

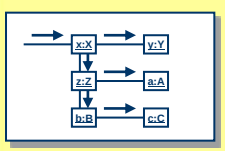




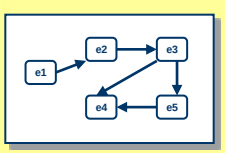
Diag. Séquence d'analyse



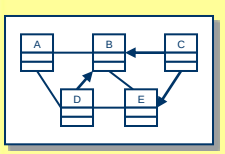
Diag. des classes participantes



Diag. de communi-cation

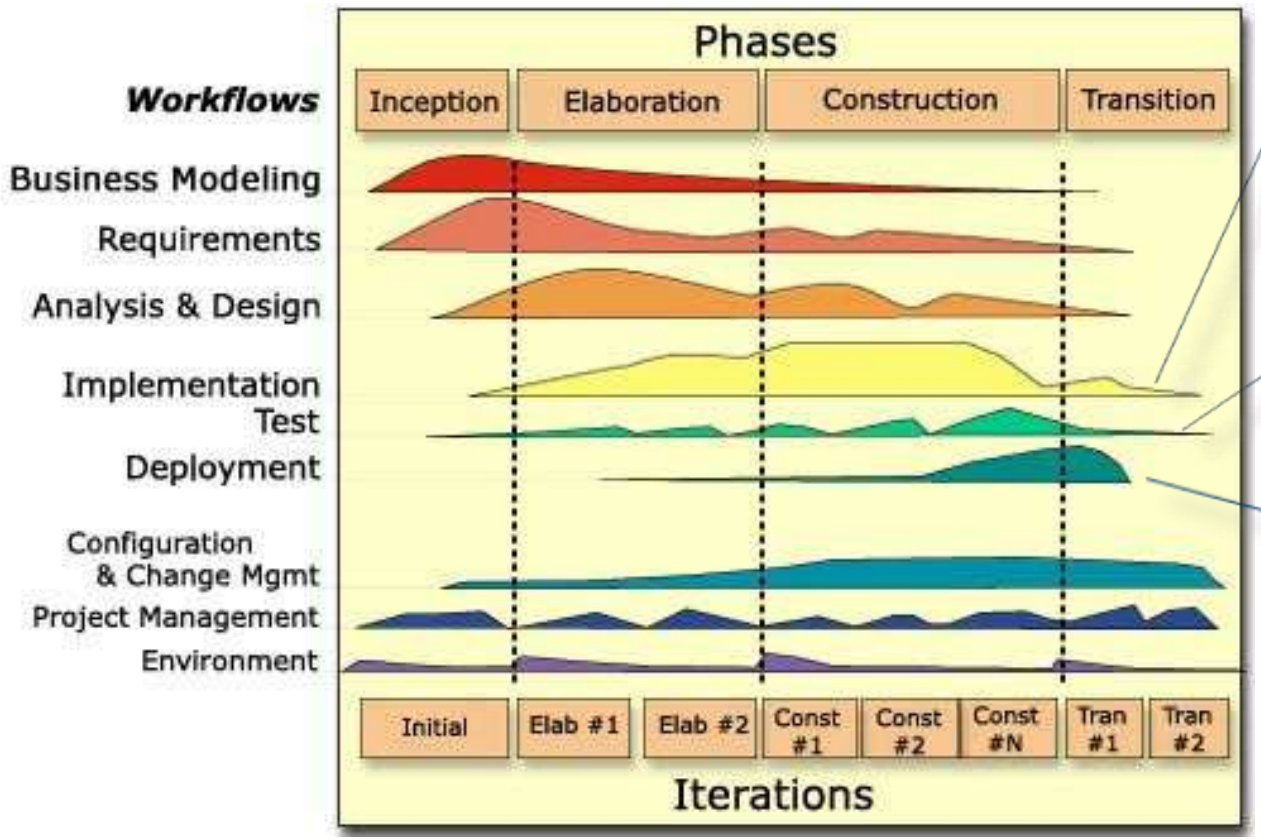


Diag. d'état-transition



Diag. des classes de conception

Principes d'organisation : les disciplines



L'agilité en douceur

- **Les points forts**
  - Très adapté aux projets importants
  - Agile sans l'être trop
    - Elle est acceptée plus facilement pour ceux qui connaissent le cycle en V
    - Structure classique de l'équipe (MOA, CP, développeurs, concepteurs, testeurs)
    - Les fondamentaux de l'agilité sont respectés
  - Permet d'utiliser des compétences différenciées
    - Concepteur, développeur, architecte, testeur
- **Les limites**
  - La moins agile des méthodes agiles
    - Elaboration d'artefact d'analyse, de conception, etc.
    - Est-ce un inconvénient ?
- **En conclusion**
  - La meilleure façon d'aborder l'agilité
  - Des équipes très autonomes préféreront un processus comme XP ou Scrum

Agenda

- Pourquoi faire de l'agilité ?
- Les piliers de l'agilité
- Unified Process
- **Extreme Programming**
- Scrum
- Les pièges et les bonnes pratiques
- Conclusion

Extreme programming

- **Genèse**
  - Kent Beck
  - Ron Jeffries
  - Première mise en œuvre chez Chrysler en 1996
  - Publié en Octobre 1999
  - Référence : « Extreme Programming Explained » (Kent Beck)
- **Quatre valeurs**
  - La communication
  - La simplicité
  - Le feedback
  - Le courage

Les rôles dans XP

Le client (ou utilisateur)

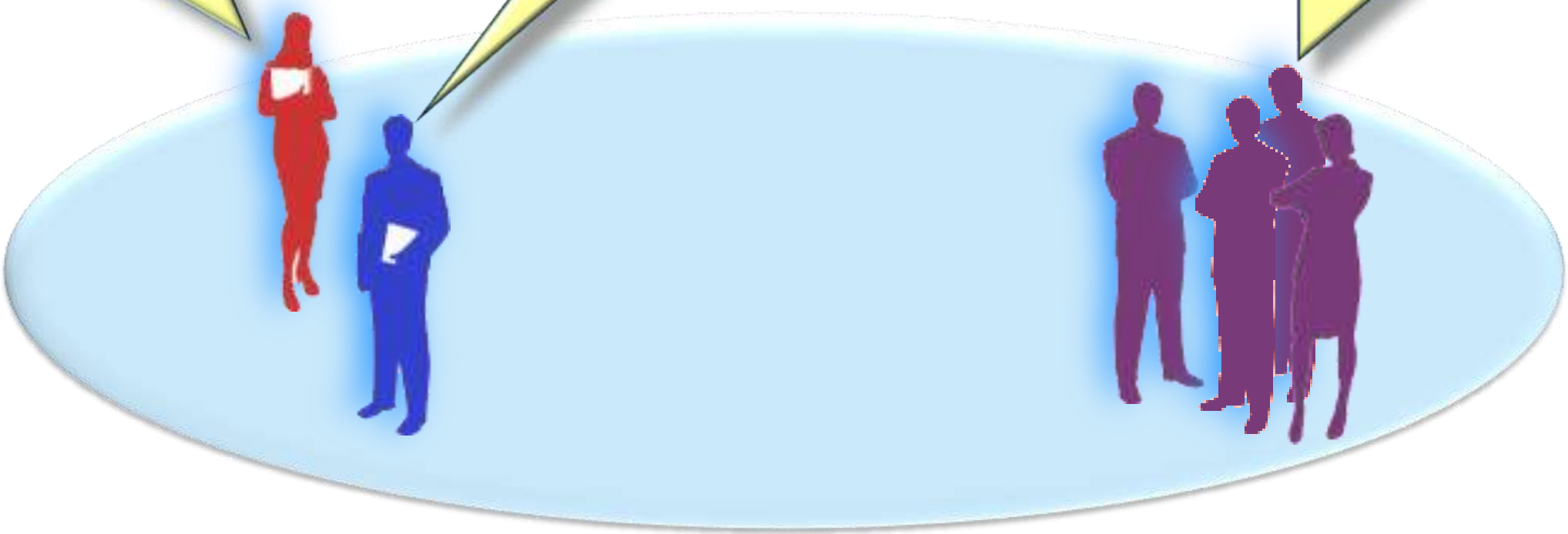
- Il spécifie les fonctionnalités à implémenter
- Il décide quelles fonctionnalités doivent être implémentées
- Il définit les tests d'acceptation
- Il décide quand le projet doit être considéré comme terminé

Le testeur

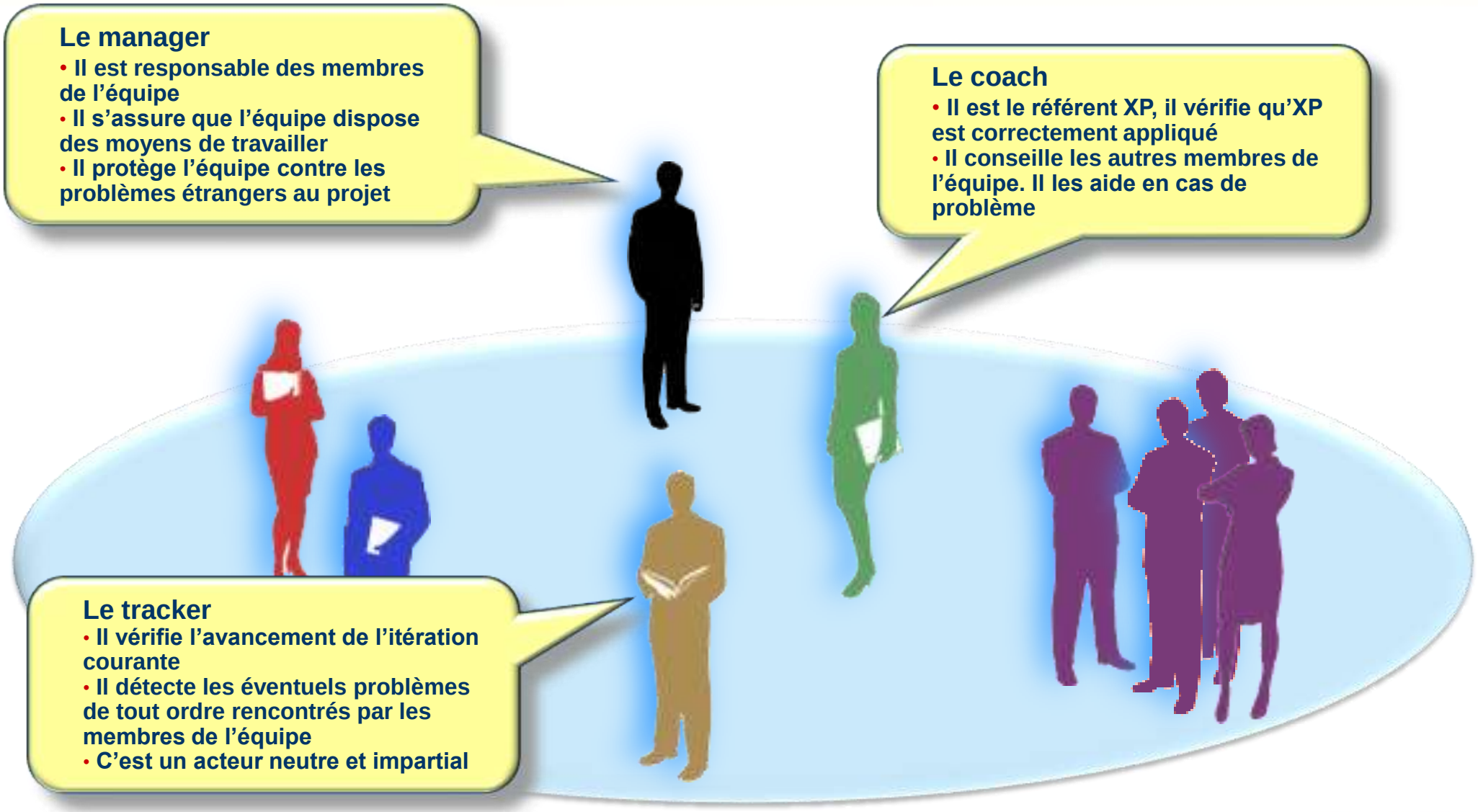
- Il implémente les tests d'acceptation définis par le client

Les développeurs

- Ils estiment la charge de travail pour chaque fonctionnalité
- Ils conçoivent et implémentent les fonctionnalités
- Ils construisent et passent les tests unitaires



Les rôles dans XP



Les bonnes pratiques XP

Client sur site !

- Pour améliorer la communication
- Pour répondre rapidement aux questions
- Pour évaluer au plus vite ce qui est développé

Développement piloté par les tests

- Le test est écrit d'abord
- Il n'existe pas de code fonctionnel sans test
- Les tests sont exécutés inlassablement

Pair programming

- On développe à deux sur un même poste
- Cela permet de partager savoir et compétences
- Cela permet de se relire immédiatement
- Cela permet d'être moins distrait (web, mail, …)

Le jeu de la planification

- Le client exprime des besoins
- Les développeurs estiment les charges
- Le client priorise les tâches à accomplir

Les bonnes pratiques XP

Refactoring
- Commencer par faire le plus simple possible
- Ne pas hésiter ensuite à réécrire du code pour l'optimiser ou le clarifier

Intégration continue
- Les développeurs livrent leur travail très fréquemment
- Le travail de tous est consolidé et vérifié régulièrement (tous les soirs)

Itérations courtes
- L'itération dure une à deux semaines

Conception simple
- Pas d'architecture ou de conception sans motivation évidente
- Implémentation la plus directe possible
- YAGNI : You Aren't Gonna Need It !

Les bonnes pratiques XP

Propriété collective du code

- N'importe quel développeur peut intervenir sur un code
- Il faut régulièrement échanger les sujets
- Cela permet de réduire l'impact des arrivées et des départs

User de la métaphore

- Une invitation à être pédagogique
- S'appuyer sur les références des interlocuteurs (utilisateurs)
- Éviter le jargon technocrate

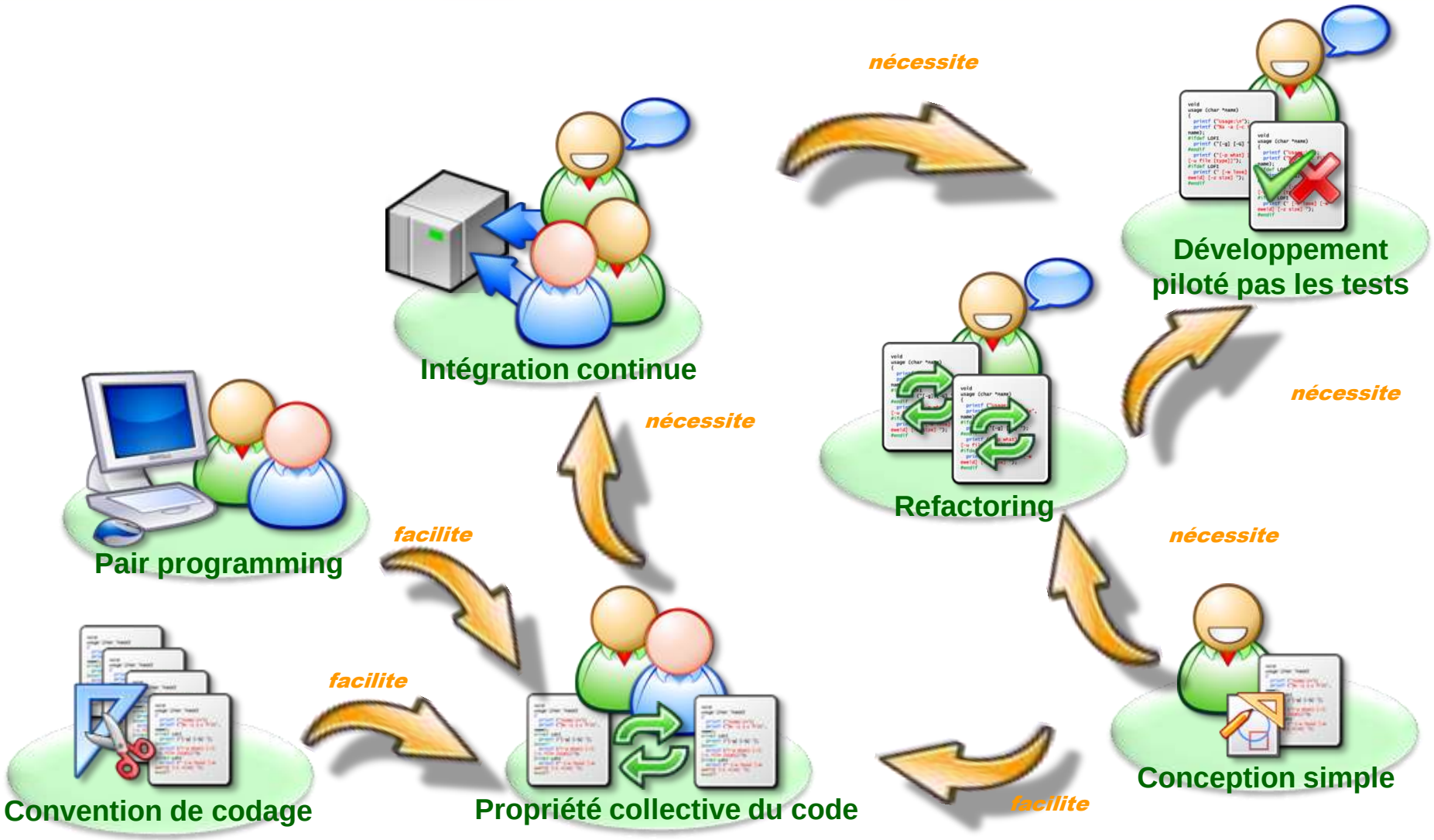
Rythme de travail continu

- Éviter les rushs en fin d'itération ou de projet
- Pas d'heure supplémentaire
- Proposer des plannings réalistes

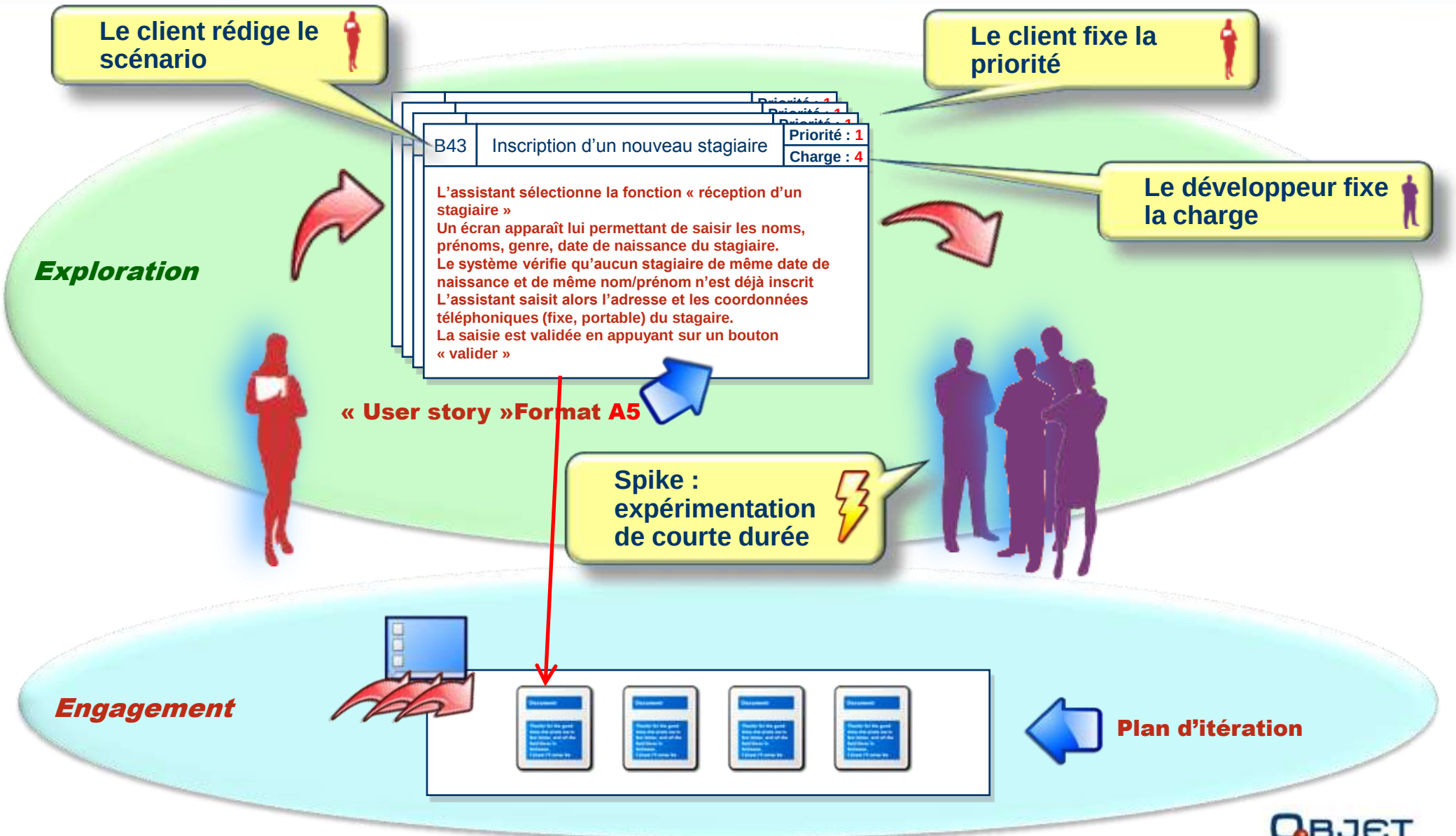
Convention de codage

- Un format partagé améliore la lisibilité
- Utiliser l'outillage (IDE, intégration continue) pour vérifier et formater

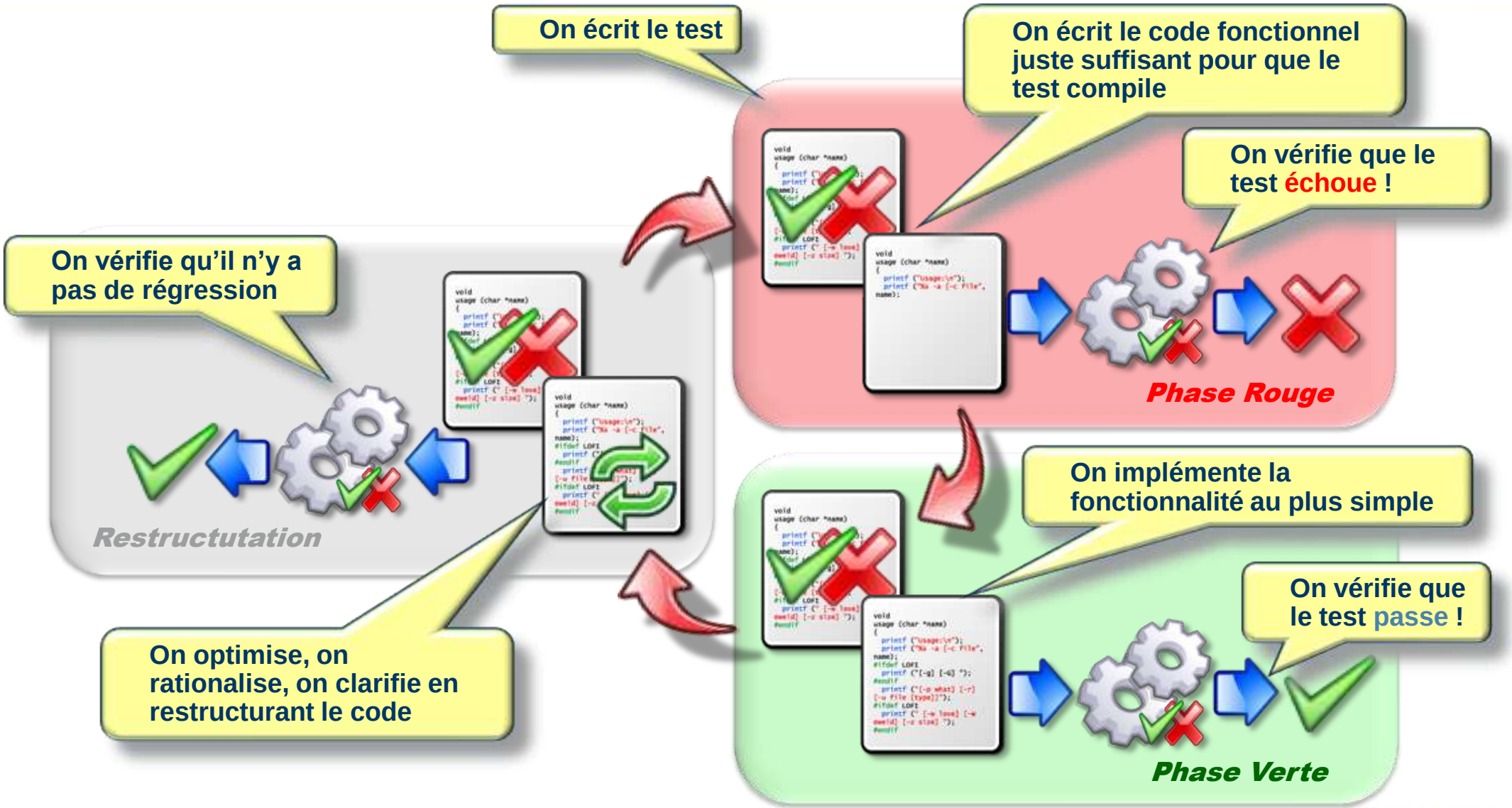
XP est un processus cohérent



Le « planning game »



Développement piloté par les tests



Intégration continue

Build local

Gestionnaire de sources

Serveur d'intégration continue

Build

Récupérer les dépendances

Compiler

Exécuter les tests unitaires

Packager

Déployer

Exécuter les tests d'intégration

Vérifier la qualité du code

Générer la documentation

Processus de build

Référentiel

Plateforme de tests

Notifications

Extrême ou original ?

- **Les points forts**
  - Produit une application de qualité (efficace, évolutive, maintenable)
  - Pas de divergence avec les exigences du client
  - Très souple face aux changements (exigence, arrivées, départs, …)
  - Évite les coûts inutiles (documentation intermédiaire, fonctionnalités jouets)
  - Garantit une grande transparence
- **Les limites**
  - Surtout adapté aux petites équipes (<12 personnes)
  - Nécessite une équipe assez homogène et de bon niveau
  - Pas adapté à un contexte forfaitaire
  - Il n'est pas toujours évident d'avoir le « client » sur site …
  - Mais ? Où est la documentation ?
- **En conclusion**
  - Très pragmatique, très efficace
  - Recommandé lorsque le contexte projet s'y prête
  - … surtout lorsque « le client ne sait pas ce qu'il veut »

Agenda

- Pourquoi faire de l'agilité ?
- Les piliers de l'agilité
- Unified Process
- Extreme Programming
- Scrum
- Les pièges et les bonnes pratiques
- Conclusion

Scrum

- **Processus de développement simple**
  - On a compris l'essentiel en une heure de présentation
- **Focalisé sur l'organisation de l'équipe et de l'activité**
  - Définit les rôles, les étapes, les réunions, les points d'avancement
  - La méthodologie de développement n'est pas abordée
  - Est souvent "complété" par XP
- **Procéssus itératif et incrémental**
  - L'itération s'appelle : le Sprint
  - Le logiciel fonctionne à la fin de chaque Sprint
  - Un statut d'avancement synthétique au quotidien

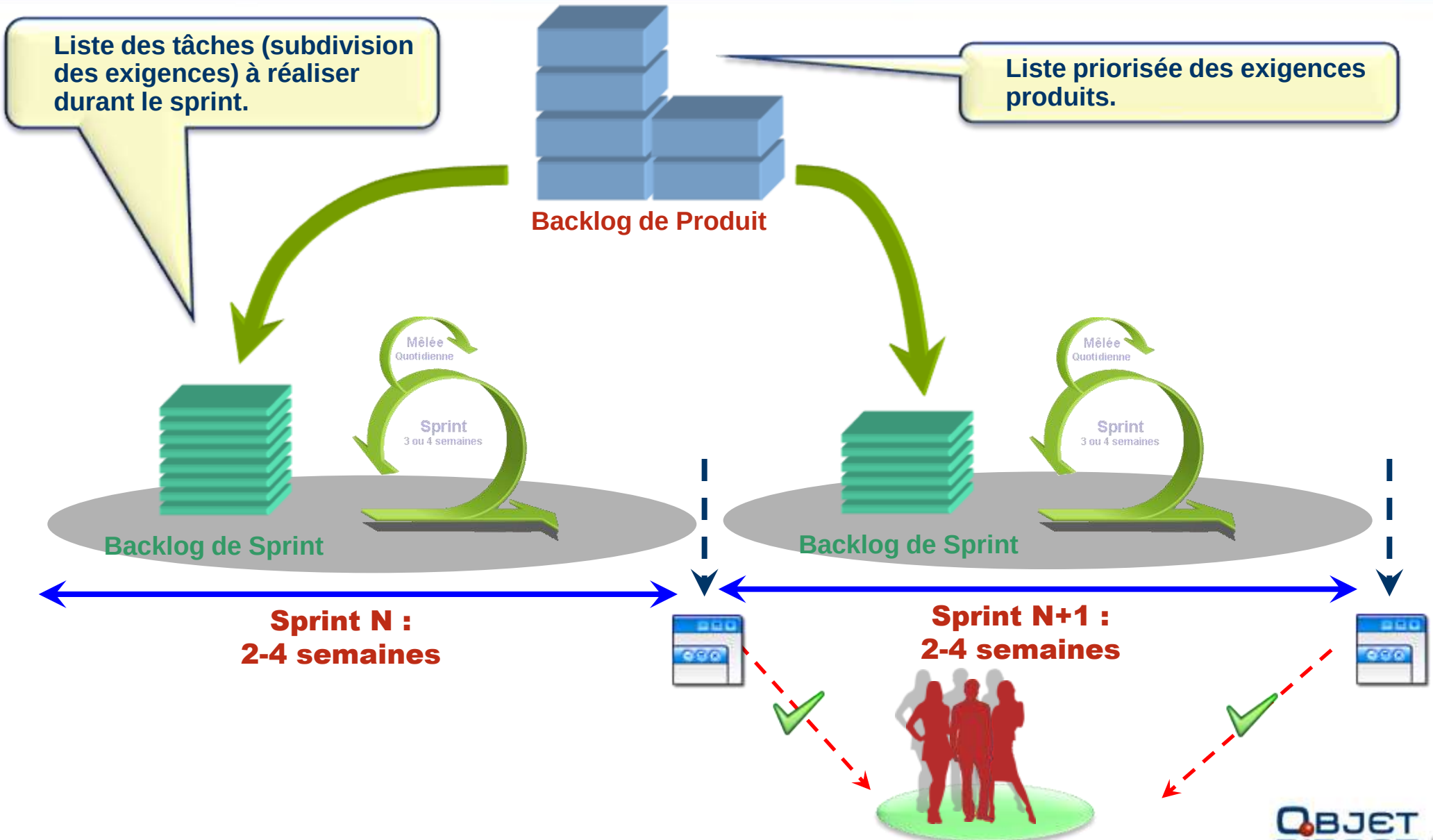
Rôles

- **Directeur Produit :**
  - Réprésente des utilisateurs.
  - Définit les fonctionnalités à implémenter
  - Priorise les demandes produits.

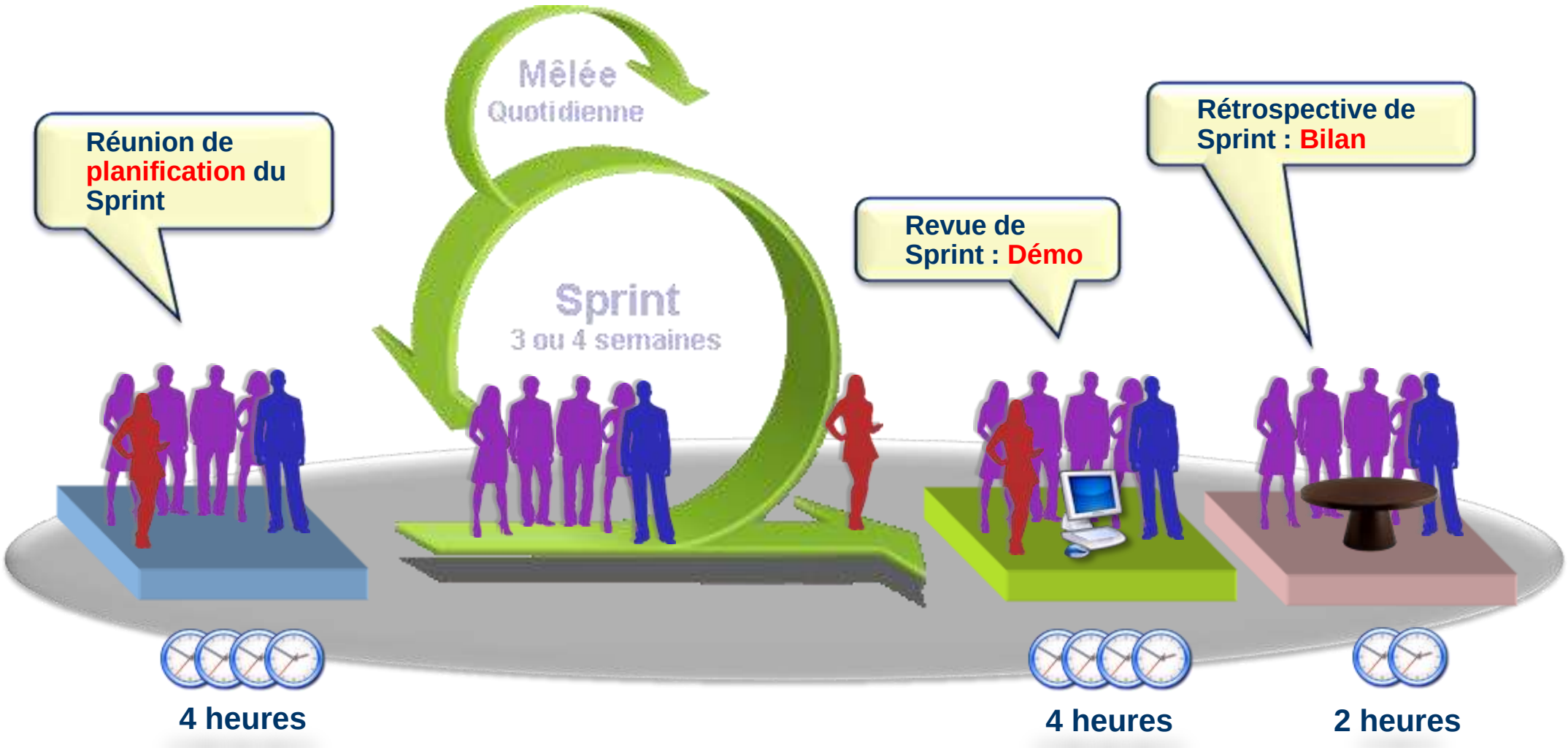
- **Equipe scrum**
  - Equipe de 5 à 10 personnes regroupant tous les rôles traditionnels
    - architecte, developpeur, testeur, administrateur
  - Effectue les estimations
  - Développe le produit
  - Se gère en toute autonomie
  - Est responsable collectivement

- **Scrum master**
  - Fait appliquer les valeurs et les pratiques de Scrum
  - Résout des problèmes d'organisation, les conflits internes
  - Supervise l'activité de l'équipe
  - Protège l'équipe de tous les élements perturbateurs externes

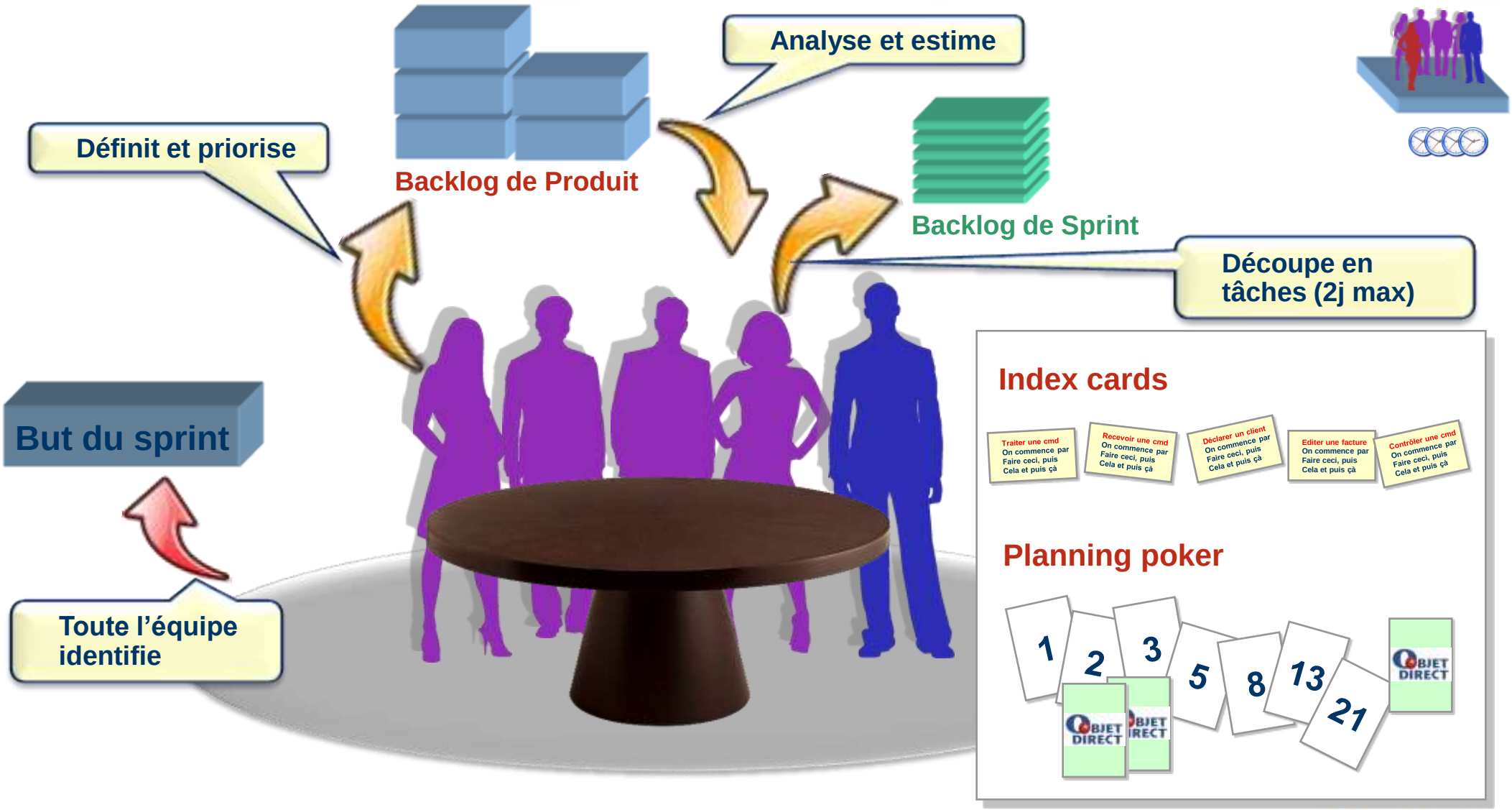
Back Log de produit et de sprint



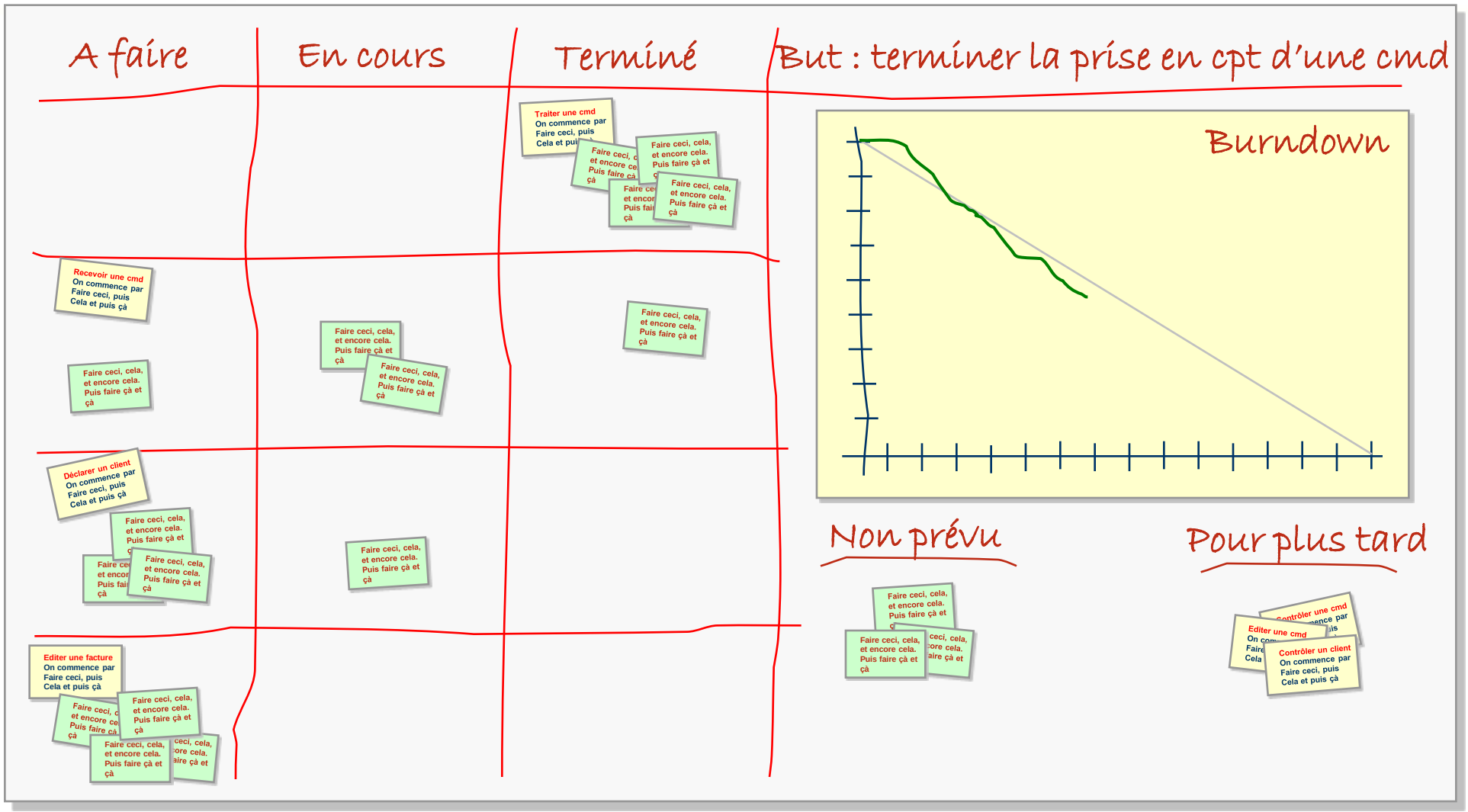
Déroulement d'un Sprint



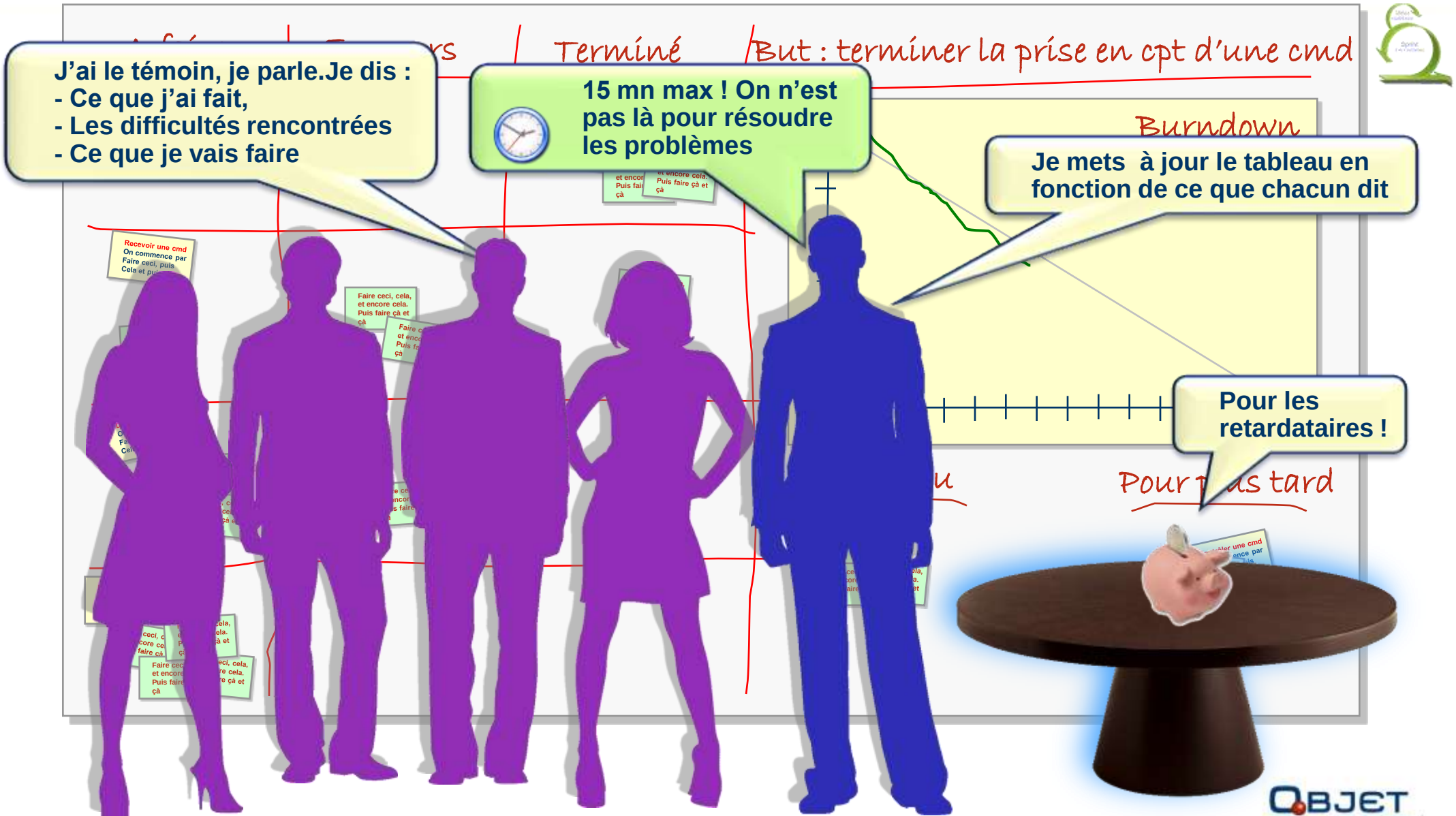
Réunion de planification d'un sprint



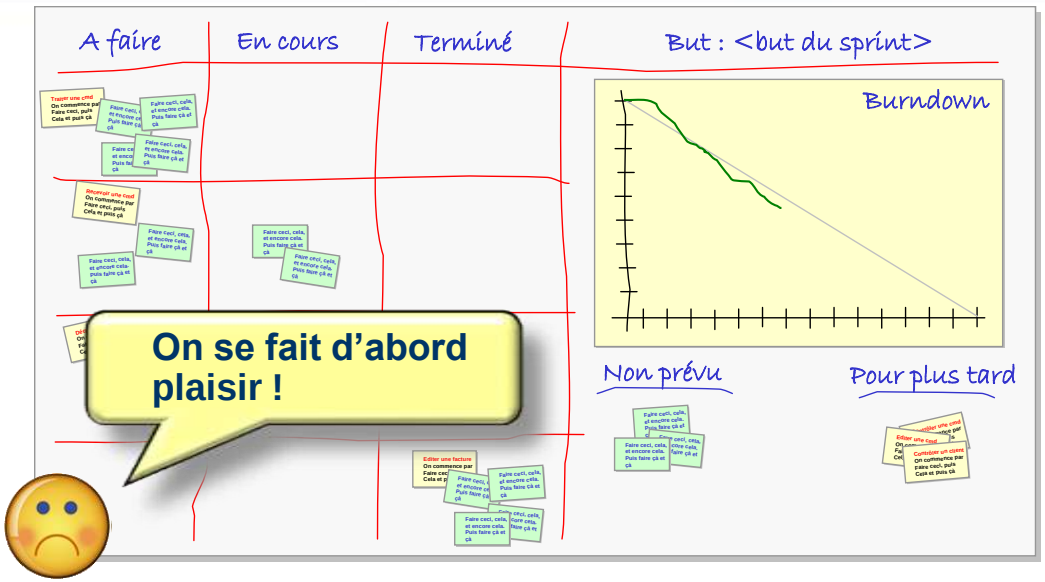
Scrum au quotidien

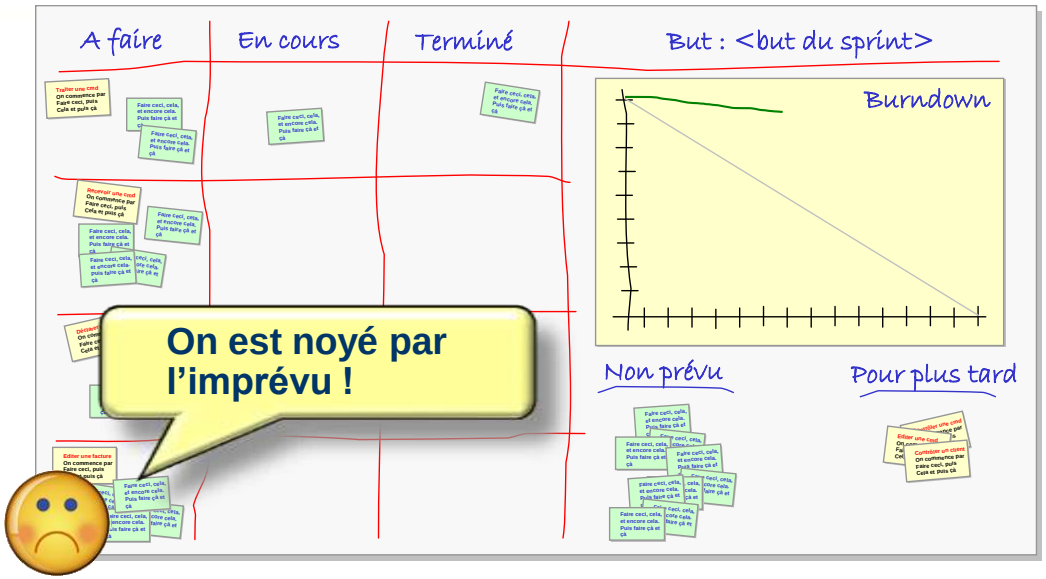


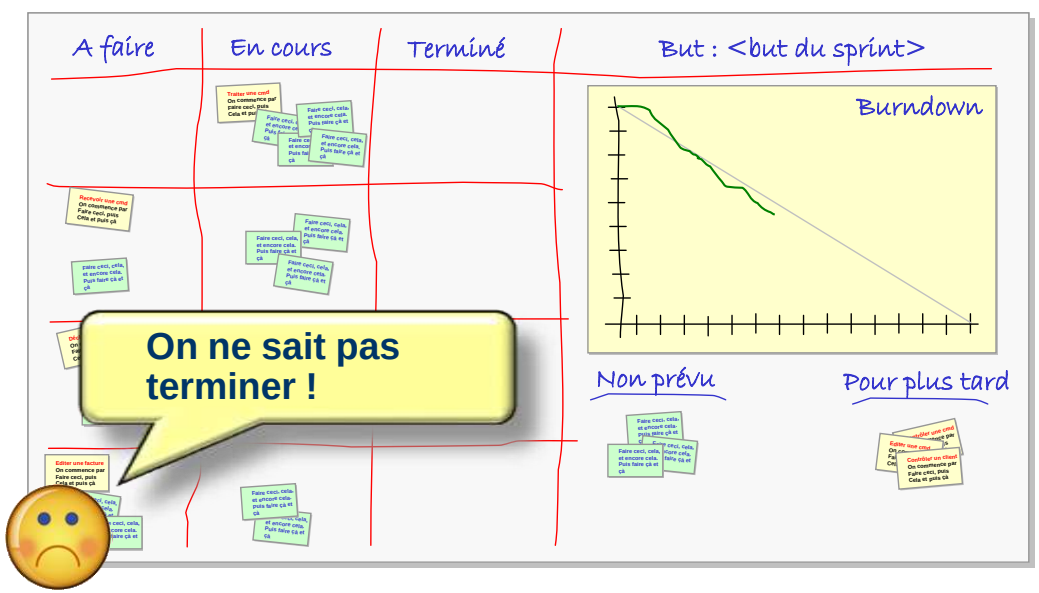
Scrum au quotidien

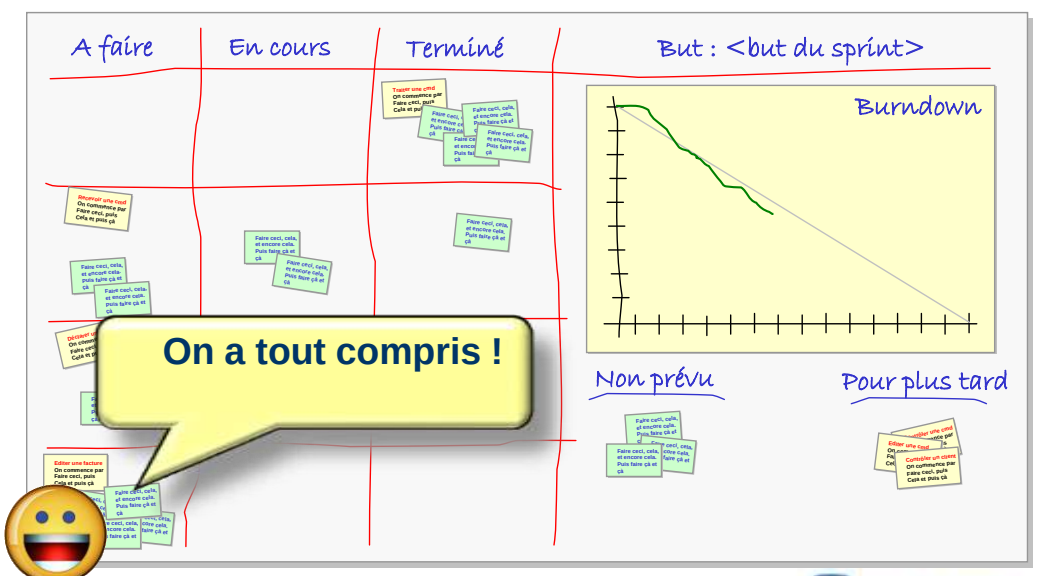


Comprendre le sprint en un clin d'oeil

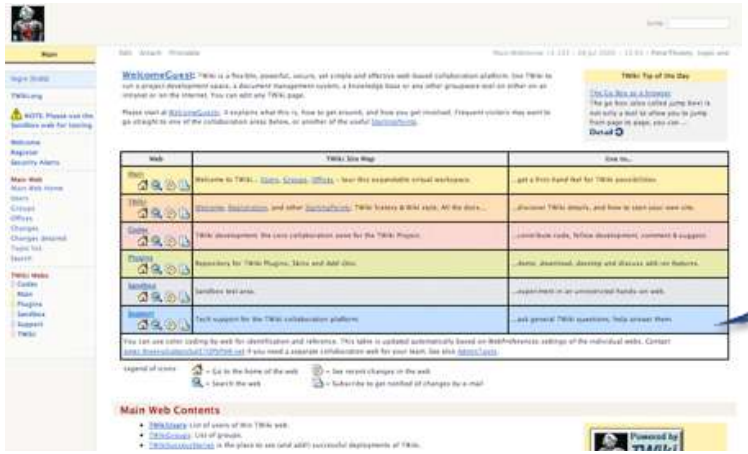




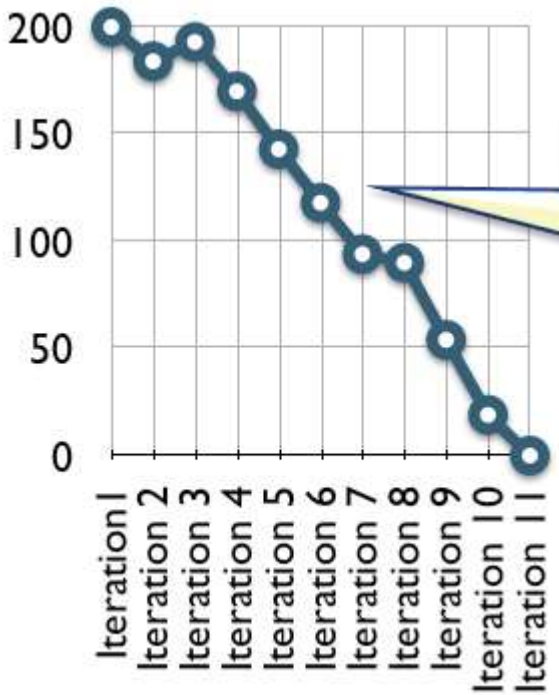




Reporting et Tableaux de bords

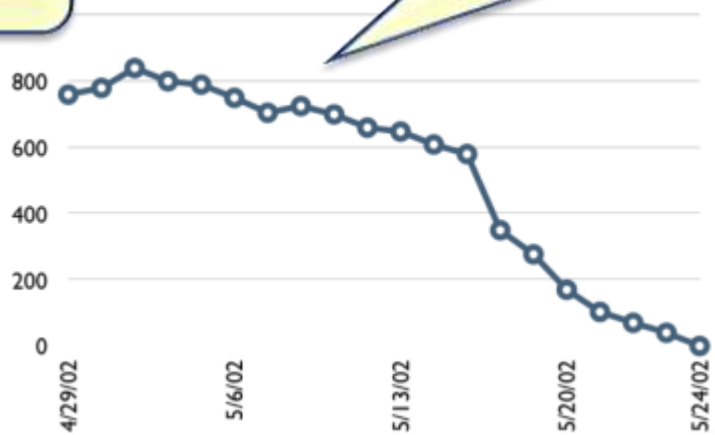


Le Wiki : pour partager la liste des exigences, des tâches, de la définition des sprints, etc.



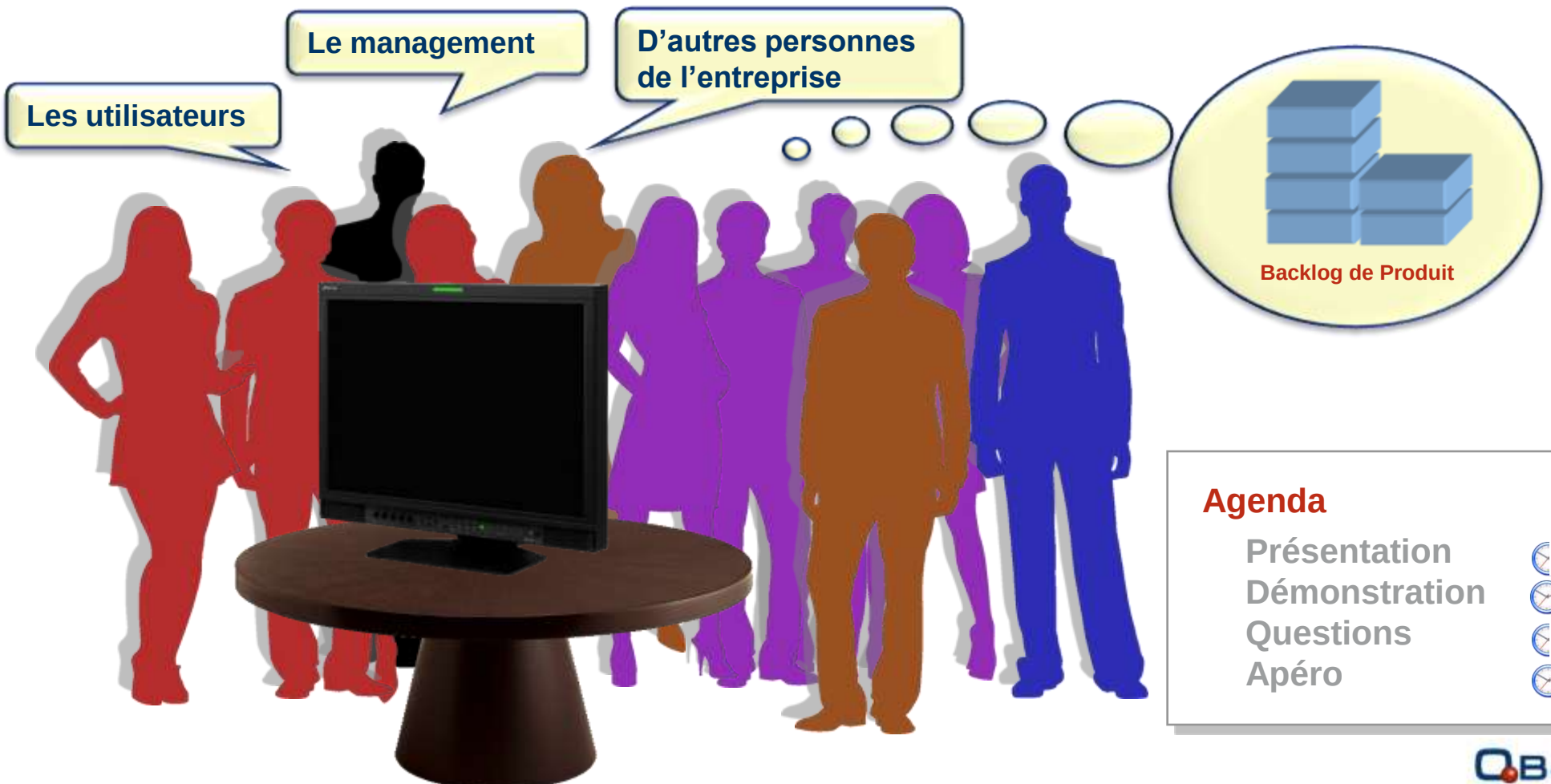
La release burndown chart : pour mesurer l'avancement du projet

Le sprint burndown chart : pour mesurer l'avancement de l'itération



Revue de sprint

- **Une démo devant tout le monde**
  - Même s'il y a peu de choses à montrer
  - Prouver l'avancement, prouver le bon fonctionnement



Agenda

- Présentation
- Démonstration
- Questions
- Apéro

Rétrospective du sprint

- **Un indicateur objectif du bon fonctionnement de l'équipe : la vélocité**
  - Nombre d'H/J "idéaux" réalisés / Nombre d'H/J consommés
  - Doit être compris entre 0,5 et 0,8 (0,7 en moyenne)
  - Si tout va bien, il doit s'améliorer avec le temps

Agenda

- Pourquoi faire de l'agilité ?
- Les piliers de l'agilité
- Unified Process
- Extreme Programming
- Scrum
- **Les pièges et les bonnes pratiques**
- Conclusion

Les pièges à éviter

- **Confondre les rôles**
  - Le coach ou Scrum master qui joue le rôle de chef de projet
  - Le client qui s'immice dans les estimations
  - L'équipe qui décide de ce qu'il faut réaliser
  - Le responsable qui affecte les tâches au lieu de laisser l'équipe s'auto-organiser
  - Mal définir la responsabilité du projet

- **Mal vendre l'agilité ou mal former les acteurs du projet**
  - Les acteurs ne savent ce que l'on attend d'eux
  - Les acteurs ne savent pas comment s'y prendre
  - Les acteurs ne perçoivent pas les gains de l'agilité

© www.123rf.com

- **Se tromper dans les objectifs de l'agilité**
  - Ne pas adapter la méthode à son contexte
  - Faire preuve d'un certain dogmatisme
  - Faire de l'agilité un but en soi

Bonnes pratiques dans la relation client⇔equipe

- Former le client pour qu'il connaisse:
  - Sa mission (rôle, objectifs)
  - Ses prérogatives
  - Le rôle et les prérogatives de l'équipe
  - Entre une demi-journée et une journée de cours/TP
  - Faire au moins un "planning game" à blanc

- Apporter du support au client:
  - Afin de l'aider à rédiger/formaliser le besoin
  - Profil assistance MOA
  - Attention ! L'AMOA ne remplace pas le client !

- Ne pas hésiter à faire des points réguliers
  - Le bon rythme: une fois par semaine
  - Avoir très vite un retour
  - Permettre au client de maturer sa réflexion pour le prochain planning game

Bonnes pratiques dans la gestion de l'équipe

- **Etre précis sur la méthode**
  - Exposer au préalable le déroulement des cérémonies
  - Bien définir ce que veut dire : "une tâche terminée"
  - Etre rigoureux sur les horaires et les rôles

- **Bien expliquer les enjeux du projet**
  - Plus-value fonctionnelle de l'application
  - Contexte économique (pour le client, pour l'équipe)
  - Contraintes et risques

- **Etre vigilant sur la gestion humaine du projet**
  - Tracer le déroulement du projet : Wiki
  - Ne pas négliger la retrospective de l'itération
  - Faire de l'amélioration un objectif majeur

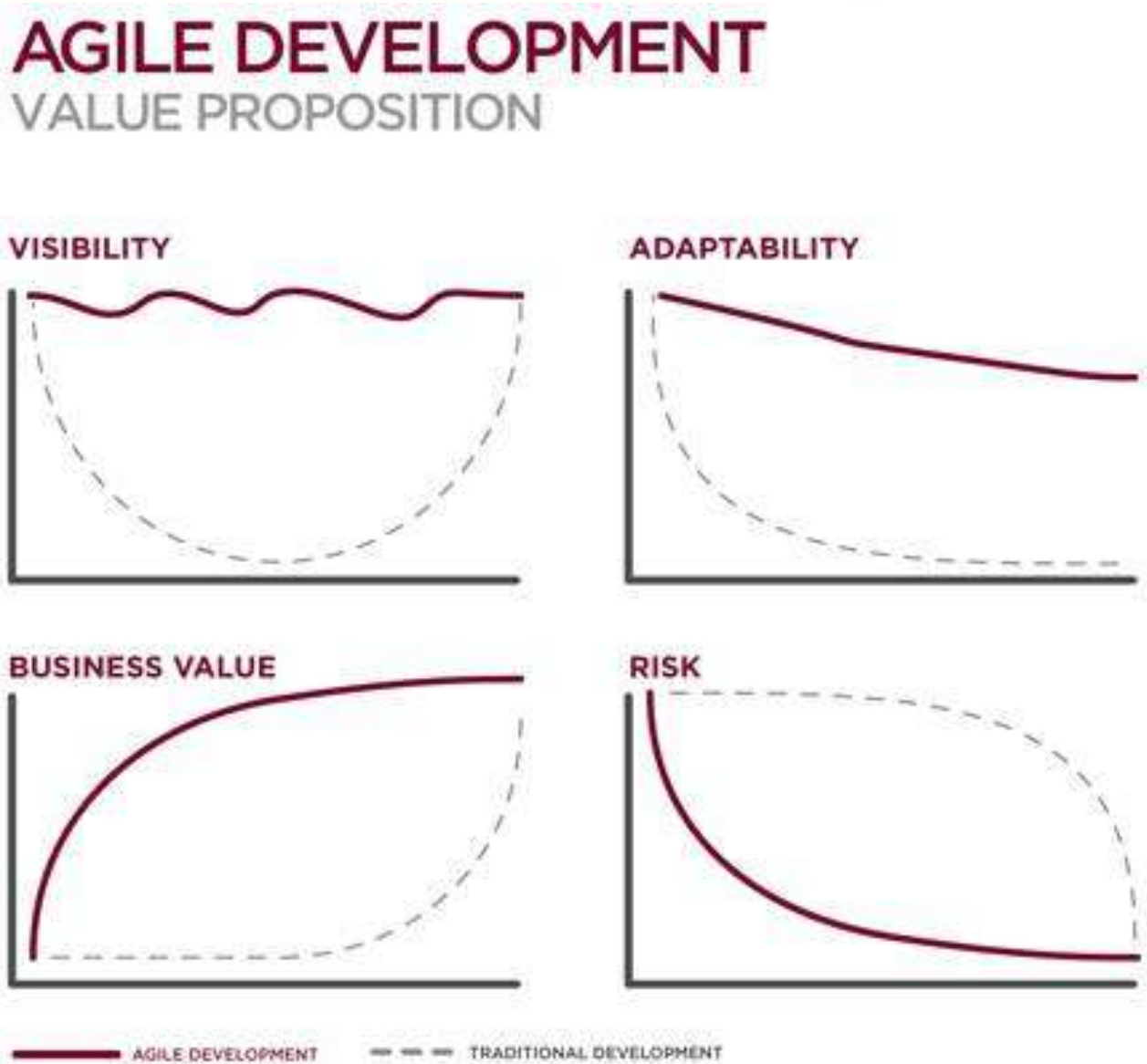
Le forfait itératif – le contrat agile

- **Au niveau projet, la SSII s'engage sur l'équipe**
  - Engagement de moyens
  - Présentation des profils affectés au projet
  - Le client s'engage sur un nombre d'itérations
- **Pour chaque itération, la SSII s'engage sur le résultat**
  - Le contenu de l'itération est estimé en "User Points"
  - Ce contenu ne change pas (SCRUM) tout au long de l'itération
  - En fin d'itération, la SSII facture au prorata du réalisé
- **Principe**
  - Elément de négociation pour le client : Il peut mettre fin à tout moment au projet, moyennant le paiement d'une soulte sur le "reste prévu"
  - Elément de négociation pour la SSII : C'est elle qui chiffre le contenu des items embarqués dans l'itération
  - Une solution médiane entre l'AT et le forfait
  - Fonctionne très bien tant qu'il existe une relation de confiance entre toutes les parties

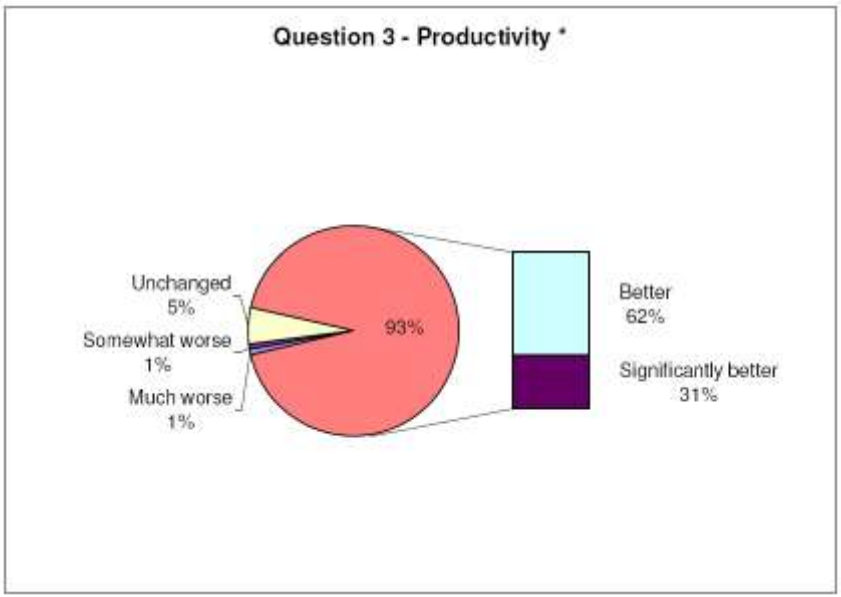
Agenda

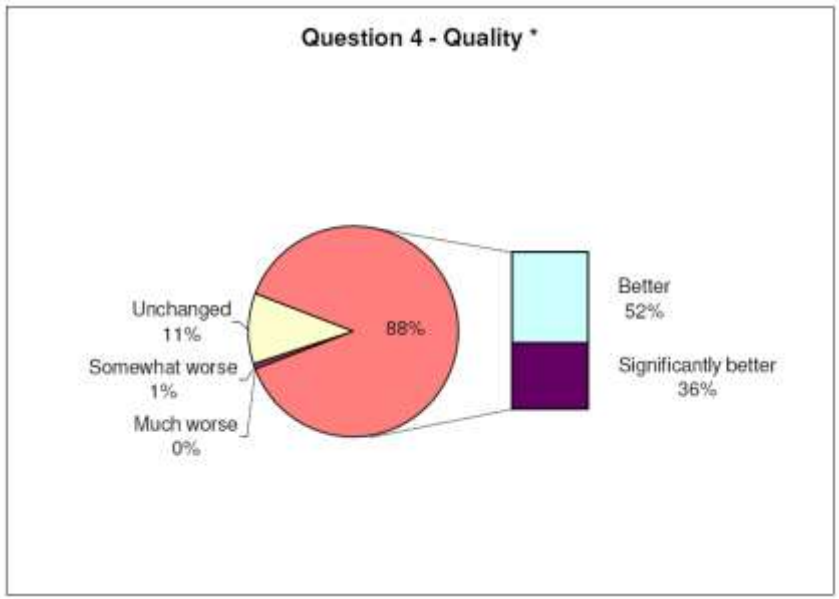
- Pourquoi faire de l'agilité ?
- Les piliers de l'agilité
- Unified Process
- Extreme Programming
- Scrum
- Les pièges et les bonnes pratiques
- Conclusion

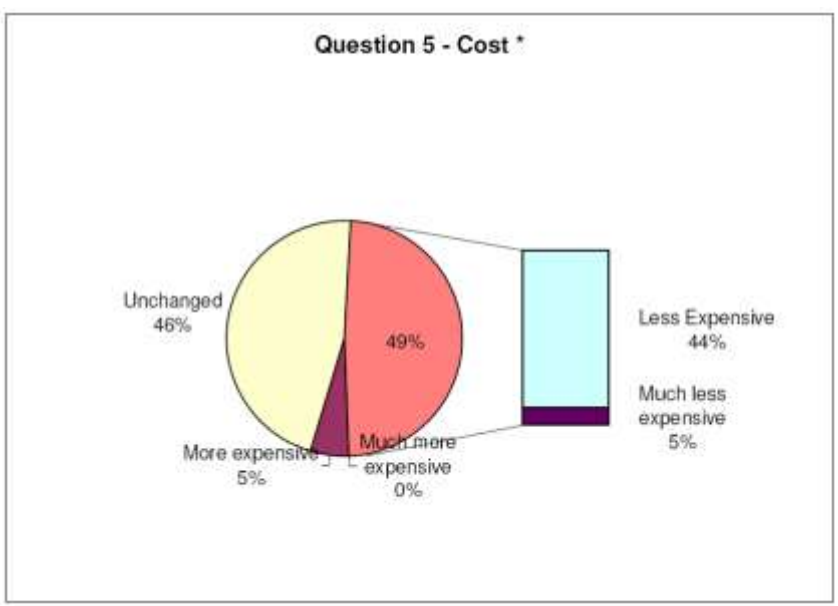
L'agilité, est-ce utile ?

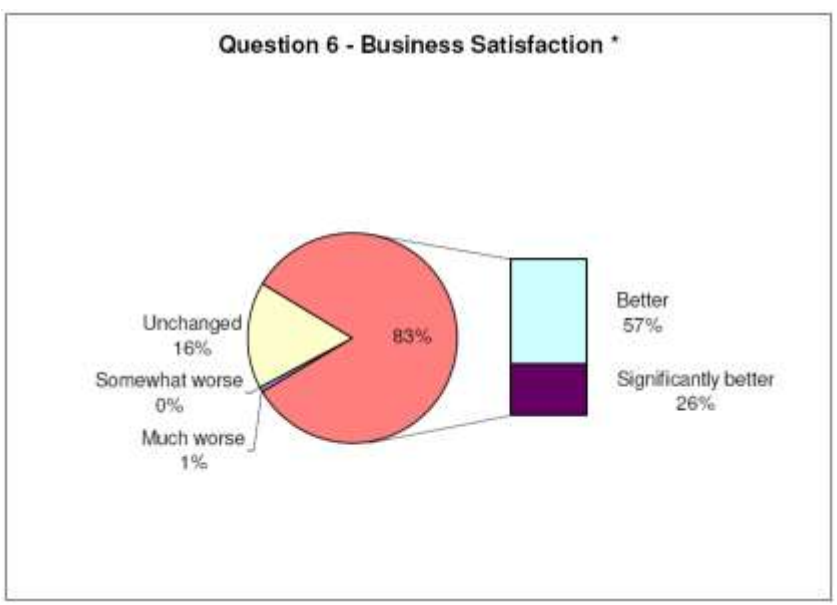


L'agilité, est-ce utile ?





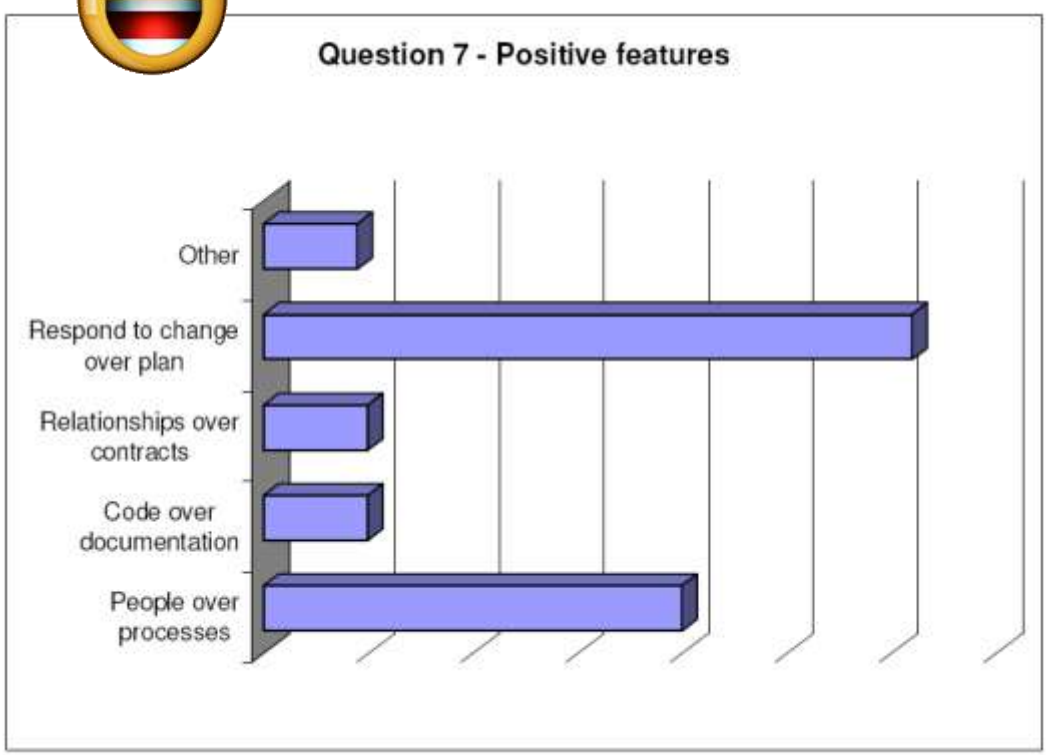


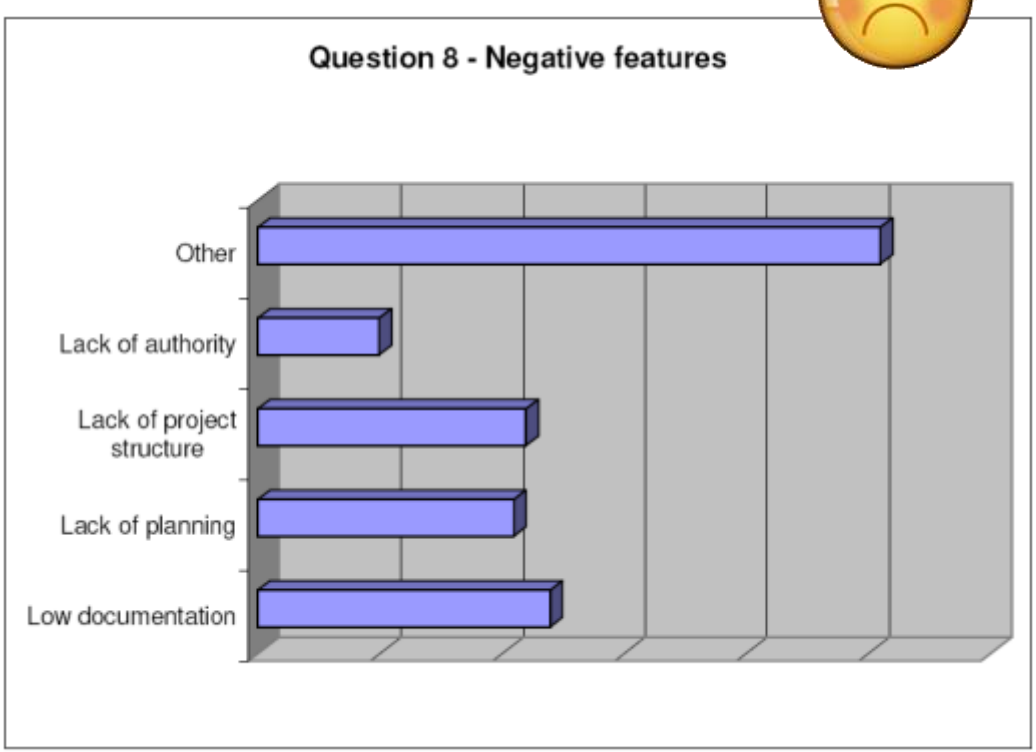


Source :
Shine Tech

Points forts et points faibles

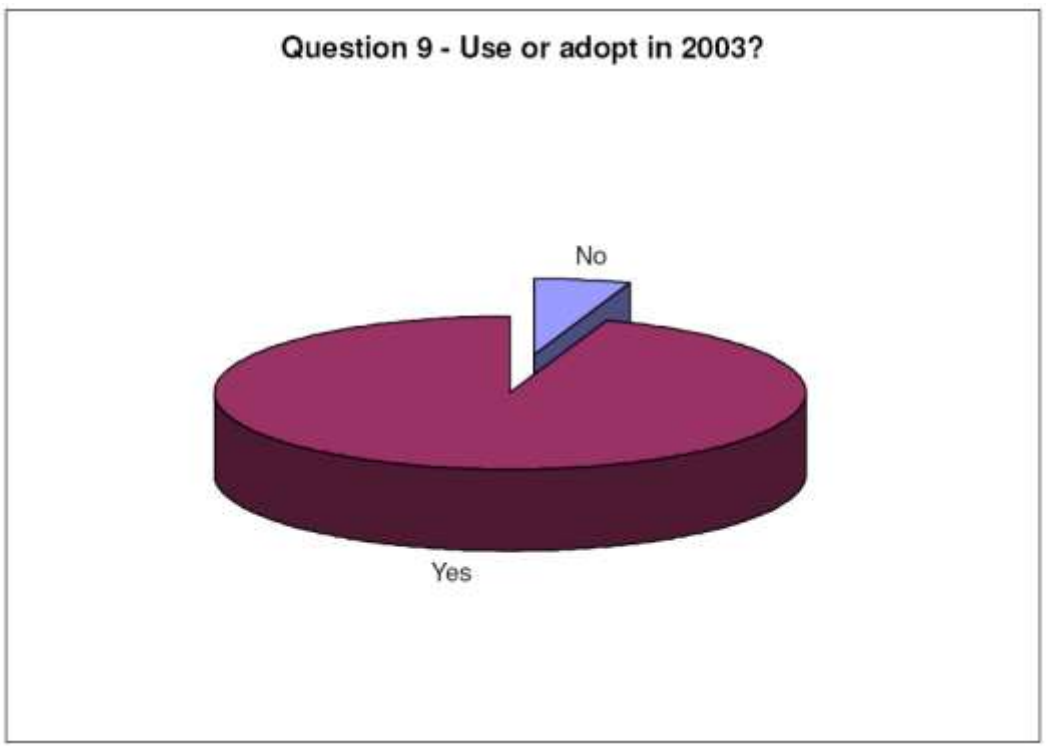
Source : Shine Tech

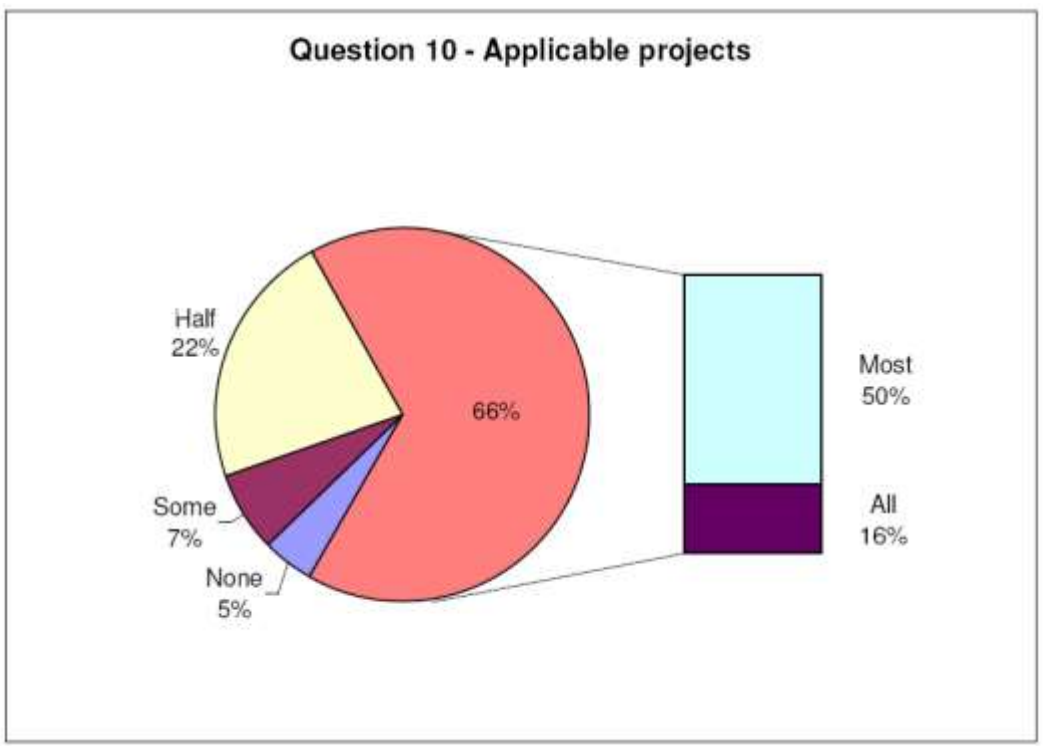




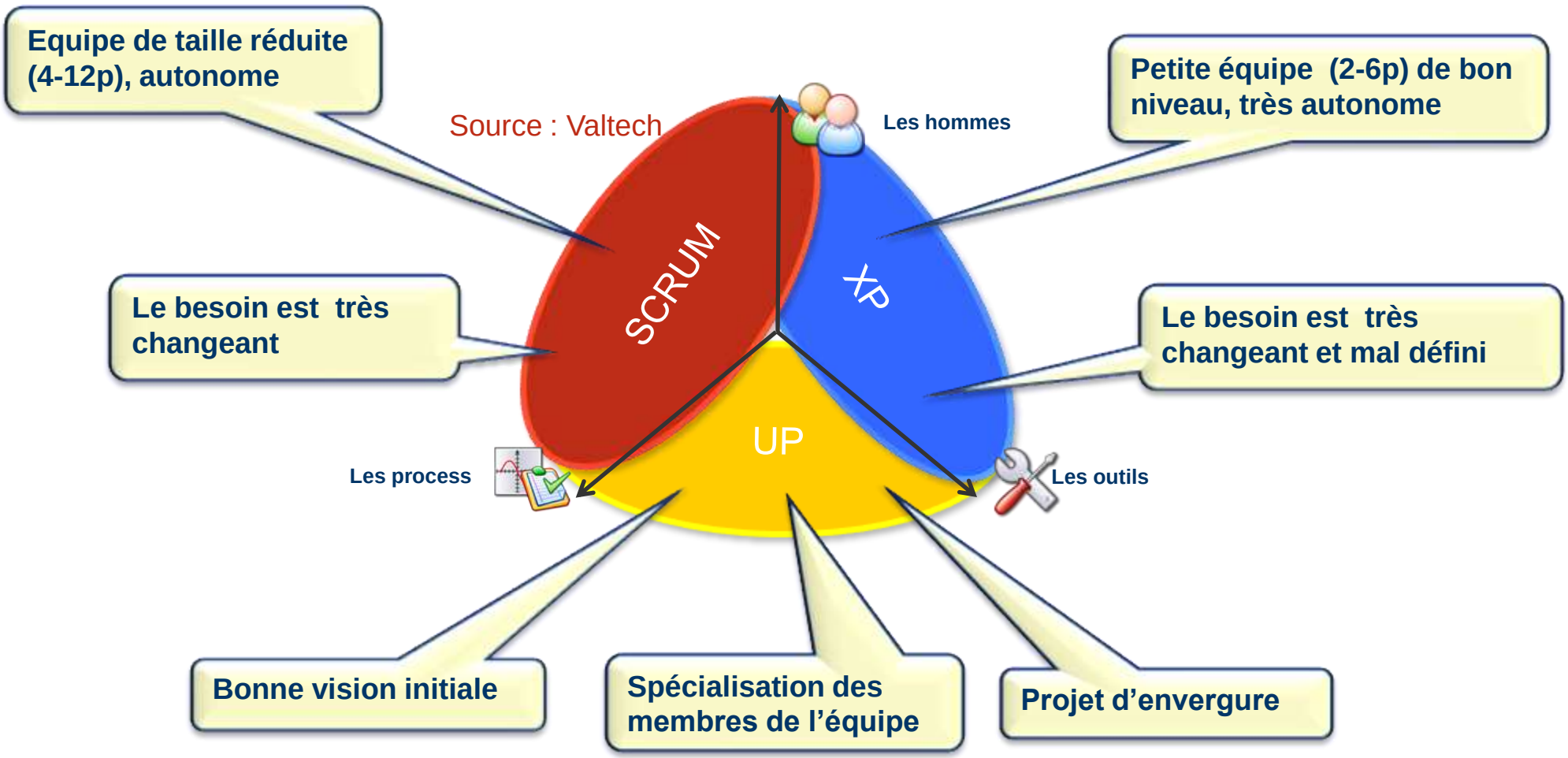
Adoption des méthodes agiles

Source : Shine Tech





Quelle méthode adopter ?

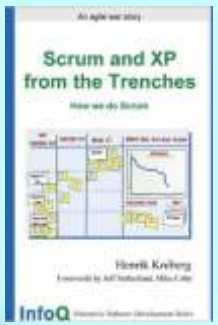


Retours d'expérience Objet Direct

- **Les MA engendrent une satisfaction générale**
  - Essayer, c'est adopter

- **Les MA limitent les tensions entre les intervenants**
  - Entre utilisateurs et développeurs
  - Entre équipes R&D, test et production
  - On prend mieux conscience du travail des autres métiers
- **Les acteurs sont davantage impliqués**
  - L'utilisateur devient un acteur écouté et responsable
  - Moins de pression s'exerce sur le chef de projet
  - Les prises de décision sont plus participatives
- **Les MA engendrent quelques difficultés**
  - Résistance au changement dans les habitudes de travail
  - Il faut savoir redéfinir les rôles de chacun …

Les livres, les blogs, les sites …



SCRUM and XP from the Trenches d'Henrik Kniberg
ISBN:978-1-4303-2264-1

-Se lit très rapidement
-Un témoignage autant qu'un manuel
-Concret, pratique, drole
-En anglais et en français
-Gratuit en téléchargement !

Le blog de ClaudeAubry: http://www.aubryconseil.com/post/Les-bénéfices-du-développement-agile

SCRUM de Claude Aubry
ISBN-10: 2100540181

- Découverte progressive
-Pratique, concret
-En français

Sur l'agilité, le site de Fabrice Aimetti: http://www.fabrice-aimetti.fr/dotclear/index.php?post/2010/05/31/L-Histoire-du-Manifeste-Agile

Les manifestations agiles

- Toutes les années en octobre-novembre
- Dans 18 villes dont 12 françaises
- Mais aussi aux USA, en Chine, au Brésil, au Québec, en Suisse, au Luxembourg
- http://www.agiletour.com
- Objet Direct est partenaire

- Connu aussi sous le nom d'XPDays
- Assez orienté XP
- Juste à Paris
- Fin mai, début juin
- http://conf.agile-france.org

N’oubliez pas la fiche d’évaluation svp