



Métodos Ágeis

- Nos dias atuais, as empresas operam em ambientes extremamente dinâmicos. Assim, precisam responder a novas oportunidades de negócios e mercados mais rapidamente.
- Novos softwares devem ser desenvolvidos rapidamente para atender as necessidades de mercados e a responder às pressões competitivas.
- Neste cenário, é praticamente impossível um conjunto completo de requisitos de software.



Os processos de desenvolvimento rápido de softwares são concebidos para produzir, rapidamente, softwares úteis. O software não é desenvolvido como uma única unidade, mas como uma série de incrementos.

- Processos são intercalados;
- Documentação minimizada;
- O sistema é desenvolvido em uma série de versões;
- Todos stakeholders;



Métodos Ágeis

- O início da década de 90, havia uma visão de que a melhor maneira para se conseguir um software de melhor qualidade era planejar cuidadosamente o projeto.
- Essa concepção vinha da comunidade de engenharia de software responsável pelo desenvolvimento de sistemas de software grandes e duradouros, como sistemas operacionais e de governo.

Na abordagem Clássica

- Abordagem mais detalhista.
- Quando aplicada aos sistemas corporativos de pequeno e médio porte, o overhead é muito grande.
- Dificuldades:
 - Muito tempo com análises;
 - Retrabalho;
 - Insatisfação;
 - Stress;



Métodos Ágeis

- A insatisfação as abordagens pesadas de engenharia de software levou um grande número de desenvolvedores a proporem o “métodos ágeis”. Estes permitiam que a equipe focasse no software em si, e não em sua concepção e documentação.
- Métodos Ágeis, universalmente, baseiam-se em uma abordagem incremental para especificação, o desenvolvimento e a entrega do software.

Abordagem ágil

equipes pequenas



mesmo espaço físico

pequenos incrementos

Po/Cliente presente

① colaboração
② negociação

desenho simples, TDD, refatoração \Rightarrow código legível

Métodos Ágeis

- Destinam-se a entregar o software rapidamente aos clientes, em funcionamento, e estes podem, em seguida, propor alterações novos requisitos a serem nas iterações posteriores.

Valores		
Interação entre Indivíduos	mais do que	Processos e Ferramentas
Produto funcionando	mais do que	Documentação Extensa
Colaboração com o cliente	mais do que	Termos Negociados (contrato)
Resposta às mudanças	mais do que	Cumprimento de planos
Ágil		Tradicional

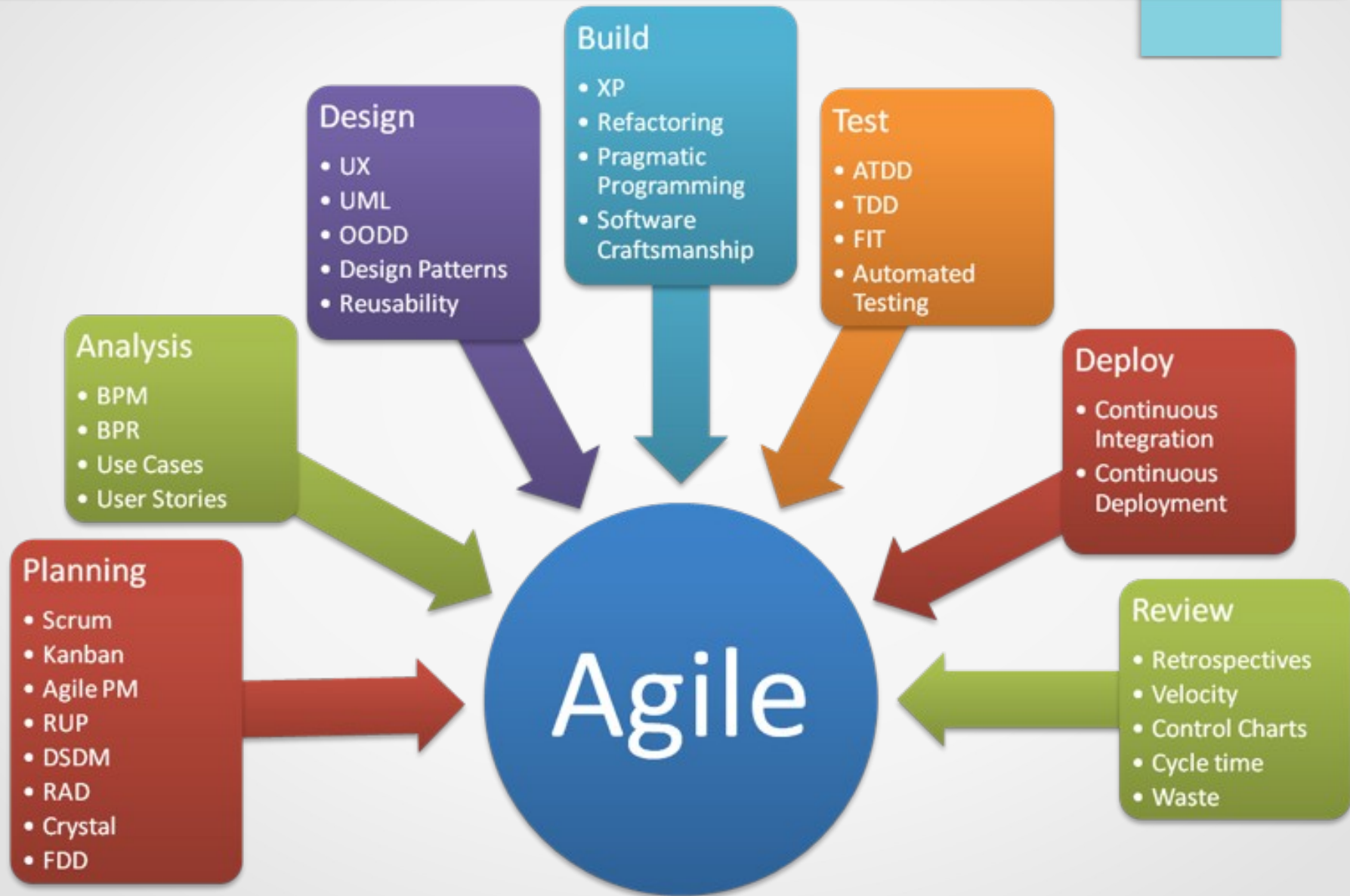
Princípios

Princípios	Descrição
Envolvimento do cliente	Clientes devem estar envolvidos no processo de desenvolvimento.
Entrega Incremental	O software é desenvolvido em incrementos com o cliente, especificando os requisitos.
Pessoas, não processos	As habilidades da equipe devem ser reconhecidas e exploradas.
Aceitar as mudanças	Projetar o sistema de maneira a acomodar mudanças.
Simplicidade	Foco na simplicidade do software quanto do processo de desenvolvimento.

Como nem tudo são flores!

Devido a seu foco em pequenas equipes bem integradas, existem problemas em escalá-los para grandes sistemas. Além disso as seguintes considerações são pertinentes:

- O sucesso depende do envolvimento dos stakeholders;
- Interação entre os membros;
- Priorizar mudanças pode ser difícil;
- Simplicidade exige trabalho extra.



Extreme Programming

Extreme Programming (XP) é reconhecida como um desenvolvimento iterativo a níveis extremos. Em XP várias versões podem ser desenvolvidas, integradas e testadas em um único dia.

- Desenvolvimento incremental sustentado por pequenos e frequentes releases;
- Engajamento contínuo;
- Programação pareada;
- Refatoração constante.

Valores, princípios e praticas

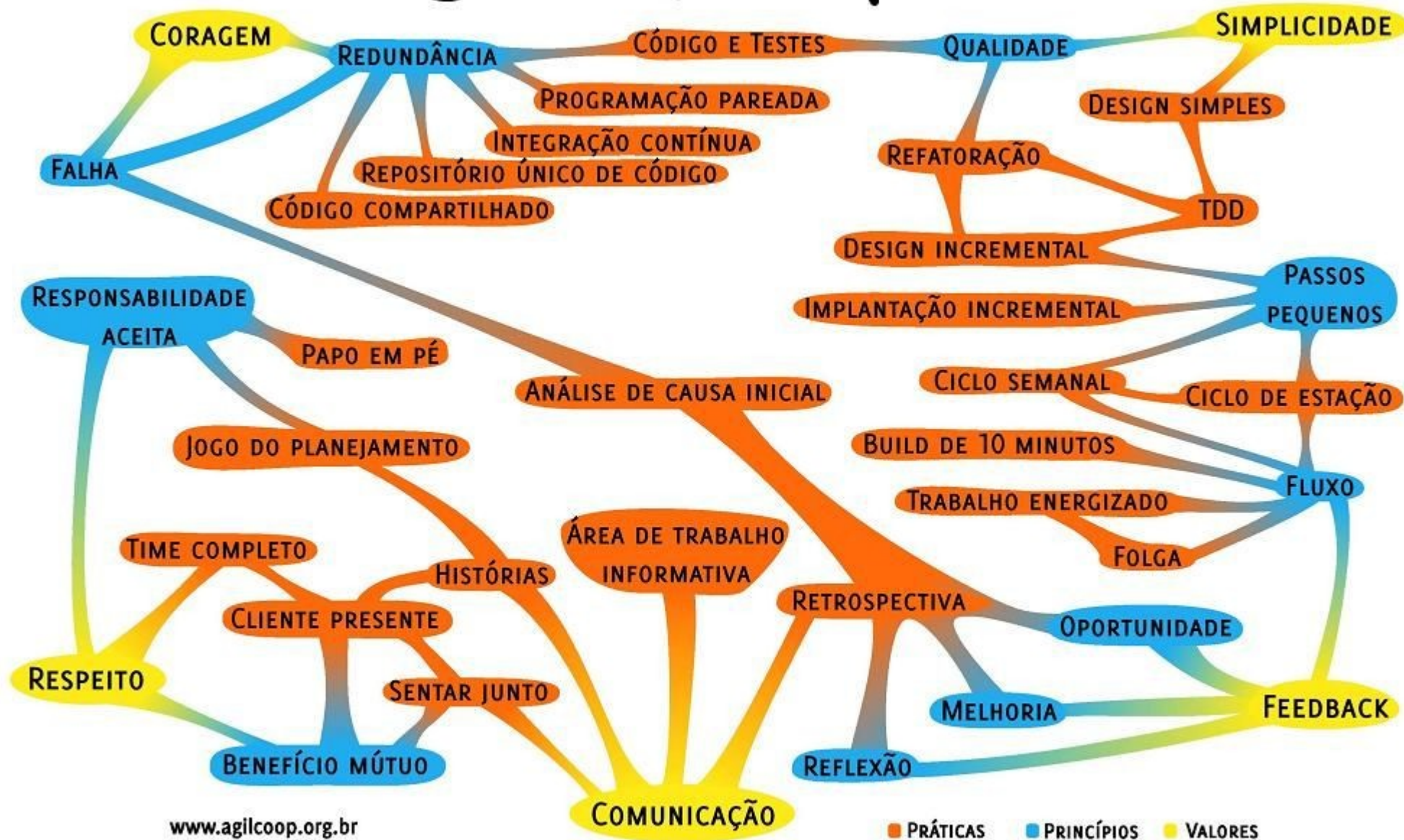
- Valores
 - Comunicação
 - Simplicidade
 - Feedback
 - Coragem
 - Respeito
- Princípios básicos
 - Feedback rápido
 - Presumir simplicidade
 - Mudanças incrementais
 - Abraçar mudanças
 - Trabalho de alta qualidade

Valores, princípios e praticas

- Praticas

- Jogo de Planejamento
- Fases pequenas
- Metáfora
- Design Simples
- Testes de Aceitação
- Semana de 40 horas
- Integração Contínua
- Propriedade Coletiva
- Programação Pareada
- Padronização do Código
- Desenvolvimento Orientado a Testes
- Refatoração

Programação eXtrema



APOIO:

LOCAWEB

REALIZAÇÃO:

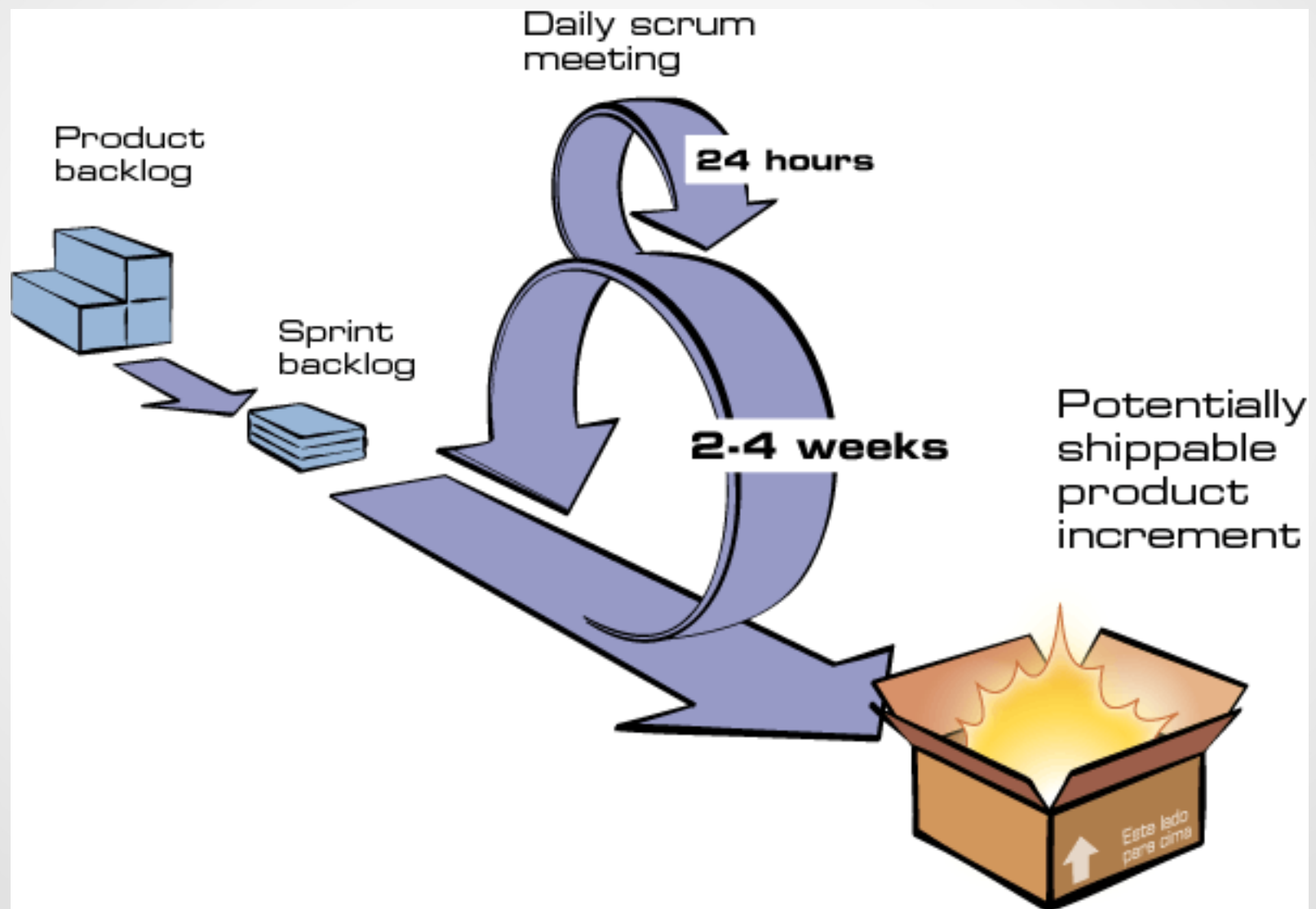
AgilCoop

Scrum



O que é Scrum?

- Scrum é uma metodologia ágil para gestão, planejamento e entrega de projetos.
- Atenção: O Scrum não é um processo padronizado onde você, metodicamente, segue etapas que vão garantir que produza um produto de alta qualidade.
- Scrum, na verdade, é um framework para organizar e gerenciar trabalhos complexos.



Práticas do Scrum



Papeis

- Dentro do SCRUM cada pessoa assume um determinado papel. A isto podemos dar o nome de representação ou função.
- Há três e somente três papeis no SCRUM: Product Owner, Scrum Master e Equipe de Desenvolvimento. Esses três em conjunto formam o time Scrum.



Product Owner



Scrum Master



Equipe de
Desenvolvimento

Product Owner

- É o principal responsável pelo produto. Tem poderes de liderança e é o único responsável por decidir quais recursos e funcionalidades serão construídos e a ordem de prioridades.
- Ele deve traduzir as necessidades de mercado ou da empresa em requisitos claros para toda a equipe.
- Além disso, ele também deve ser capaz de priorizar as necessidades de maneira que o ROI seja maximizado.

Scrum Master

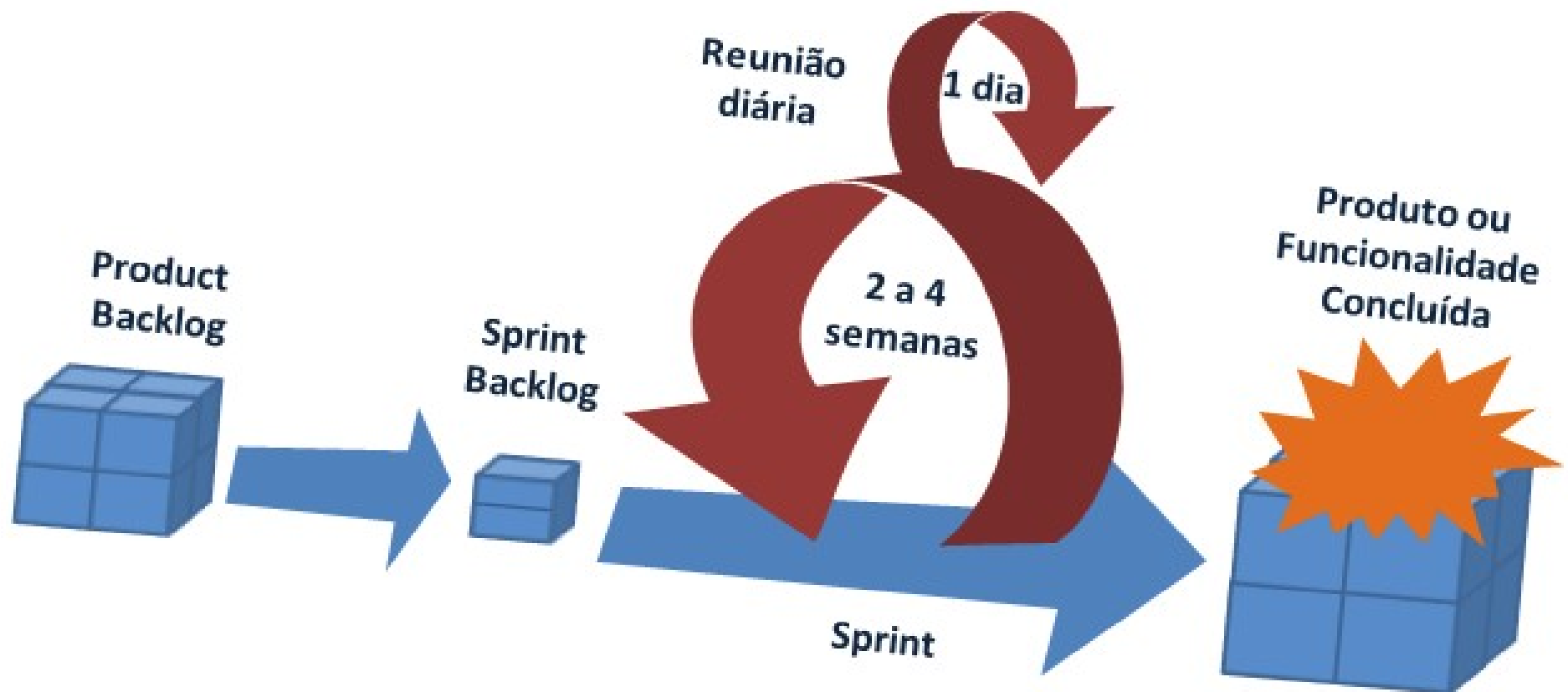
- Responsável por remover barreiras, ajudar a todos os envolvidos e garantir que o SCRUM funcione de forma efetiva.
- Age como um treinador e líder técnico e trabalha para que a equipe tenha a melhor performance possível.
 - Protege a equipe;
 - Trabalha como facilitador;
 - Atua como gerente de projeto;
 - Ajuda pessoas;

Time

- No SCRUM é definido o papel do Time de Desenvolvimento, que é simplesmente a junção de todas as pessoas em uma equipe multidisciplinar.
- Os desenvolvedores são auto-organizáveis e possuem múltiplos conhecimentos. Não há programadores, designers, DBA, etc. Todos são desenvolvedores.

OBS: Todos os membros do time devem ter coletivamente todas as habilidades.

Processos, atividades e artefatos



Artefatos

- O Product Owner tem a visão geral do projeto. Todas as necessidades são divididas em um conjunto de funcionalidades que são compiladas em uma única lista chamada de Product Backlog.
 - Definição de itens;
 - Sequenciais;
 - Itens de maior valor no topo;
 - Constante evolução;



O Product Owner é o “gerente” da lista.

Artefatos

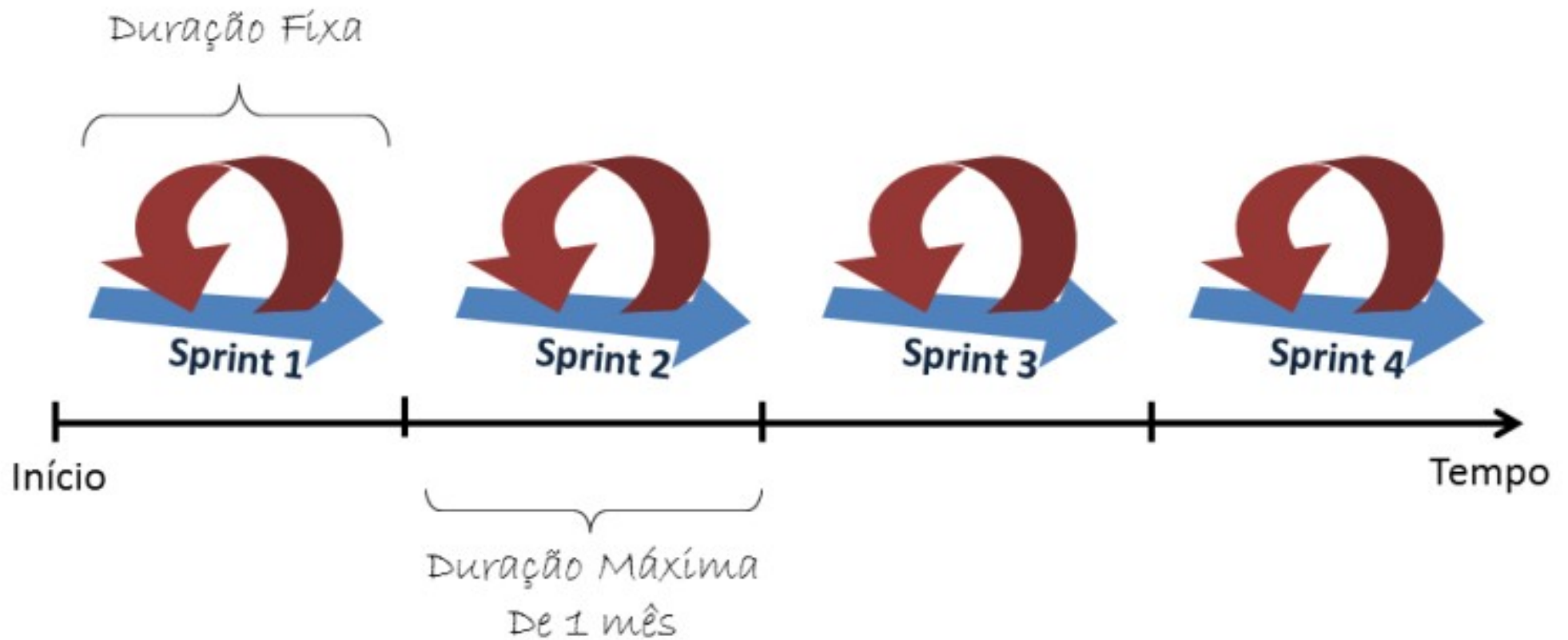
- A equipe TODA se reúne e decide a força necessária para se concluir os itens listados.
 - Cada item é “custeado” ou seja, a equipe irá mensurar quanto tempo será gasto para desenvolver cada item.
-
- Jogo de cartas;
 - Todos participam;
 - Pode ser em horas/dificuldade;



Processos

- No SCRUM, o trabalho é realizado em iterações ou ciclos de no máximo um mês de duração.
- Antes de se iniciarem o desenvolvimento dos itens todos se reúnem no que é chamado de Sprint Planning Meeting.
- Product Owner, Scrum Master e Time discutem quais são as atividades serão desenvolvidas e a equipe deixa claro o quanto consegue se comprometer.

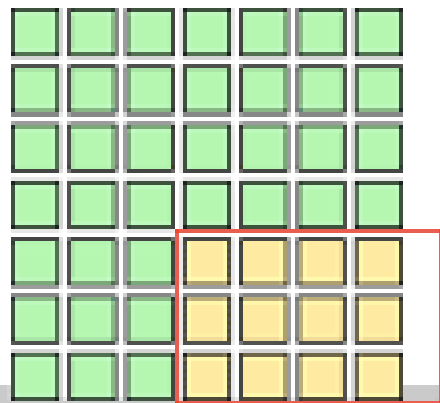
Sprints



Sprints

- Cada Sprint é um ciclo de trabalho, uma iteração. O trabalho deverá refletir um valor tangível e útil para o cliente.
- Um novo Sprint se inicia imediatamente após a conclusão do anterior, sempre realizando um Sprint Planning Meeting antes do início dos trabalhos.

Product Backlog



Sprint Backlog



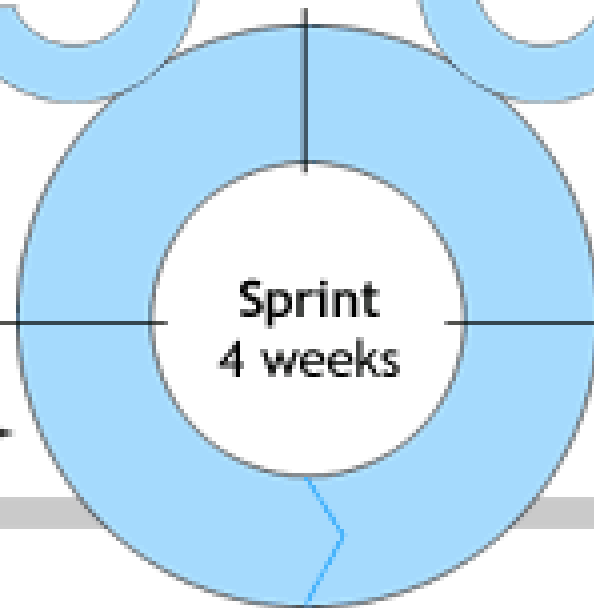
Daily meetings



Sprint burn down



Sprint
4 weeks



Sprint review
meeting



Release



Daily SCRUM

Todos os dias os membros da equipe de desenvolvimento devem realizar uma reunião com tempo definido. Esta reunião de equipe é chamada de Daily Scrum.

Durante a reunião o SCRUM Master questiona:

- O que eu fiz ontem?
- O que eu vou fazer hoje?
- Tenho problemas para desenvolver meu trabalho?

Sprint Review

- Ao final de cada Sprint é feito um Sprint Review. Durante esta reunião, o time mostra o que foi desenvolvido através de uma demo funcional.
- Product Owner, Scrum Master, gerente, cliente, etc participam da reunião e podem sugerir mudanças melhorias.
- Ao final, o Product Owner diz se aceita ou não as funcionalidades desenvolvidas no Sprint.

