

# 1

## Introdução

Projetos de software normalmente estão bastante suscetíveis a passar por inúmeras modificações ao longo do seu ciclo de vida. Muitos deles falham ao atingir seus resultados necessários dentro do custo e do prazo planejados (Matiscola 2007). O desenvolvimento de projetos demanda um esforço singular que envolve lidar com atividades, recursos e fluxo de trabalho, dentre outros elementos, para se alcançar os resultados almejados. O gerenciamento de projetos é uma questão fundamental para o sucesso do desenvolvimento de sistemas de software e tem como objetivo principal identificar e manter os riscos de fracasso em um nível tão baixo quanto necessário durante o ciclo de vida de um projeto. Exemplos de atividades essenciais e não triviais realizadas no gerenciamento de projetos são: (i) estimação de prazo e custos, a qual é diretamente afetada pelas características do projeto, quantidade de recursos alocados e pelo escalonamento de suas atividades, dentre outros; e (ii) replanejamento de atividades e realocação de recursos, dado que projetos de software tipicamente passam por diversas modificações ao longo do seu ciclo de vida, essas duas atividades tornam-se questões críticas. Além disso, quanto maior o número de projetos envolvidos, maior a complexidade no gerenciamento dos mesmos (Schwalbe 2002).

Mesmo quando gerentes de projeto têm um bom entendimento acerca das dificuldades existentes em um projeto, eles frequentemente enfrentam problemas para estimar e justificar o custo e tempo para corrigir suas causas (Matiscola 2007). O gerenciamento de projetos pode ser bastante facilitado pela utilização de ferramentas (Joslin e Poole 2005), diretrizes, modelos, sistemas de suporte a decisão automatizados e semi-automatizados ou quaisquer outros meios que auxiliem no gerenciamento das decisões que envolvem recursos, atividades e o fluxo de trabalho ao longo da execução de um ou mais projetos simultaneamente (Callegari e Bastos 2008). Além de tais decisões internas, é extremamente desejável que tais ferramentas levem em consideração o

tamanho dos mesmos, uma vez que o esforço demandado para gerir projetos de pequeno porte não é o mesmo esforço para gerir os de médio e grande porte.

Construir software de grande qualidade é uma tarefa complexa, pois os mesmos possuem diversas partes e têm muitas interações. O ramo da Engenharia de Software tem o papel de prover estruturas e técnicas que diminuam essa complexidade. Nesse contexto, duas tecnologias promissoras foram recentemente integradas (Nunes et al. 2009a), Linhas de Produto de Software (LPSs) e Sistemas Multi-agentes (SMAs), as quais apresentam características que potencialmente podem beneficiar o desenvolvimento de ferramentas de gerenciamento de projetos. LPSs (Clements e Northrop 2002) é uma abordagem interessante no ponto de vista do reuso de software que permite a construção sistemática de famílias de aplicações a partir da exploração das suas partes em comum. Já SMAs (Wooldridge 2001) são sistemas baseados no conceito de agentes, os quais agregam características como pró-atividade, autonomia e sensibilidade ao contexto. Adicionalmente, agentes podem apresentar racionalidade, aprendizado e adaptação.

Assim, nessa dissertação propõe-se explorar os benefícios de SMAs e LPSs para construir ferramentas que auxiliem no gerenciamento de projetos software que seguem a metodologia de desenvolvimento do Rational Unified Process (RUP) (Kruchten 1999), metodologia utilizada no setor de desenvolvimento de sistemas da Petrobras na Bacia de Campos. A referida equipe atuou em parceria para o desenvolvimento deste trabalho. Apesar do trabalho ser baseado no RUP, o mesmo pode ser customizado para trabalhar com outras metodologias. A idéia principal é o desenvolvimento de uma linha de produto de sistemas de gerenciamento de projetos de software baseados em agentes. Tendo em vista que o gerenciamento de projetos pode variar de acordo com o tamanho do projeto, visa-se desenvolver uma arquitetura de LPS a fim de possibilitar a derivação sistemas customizados que sejam adequados para o tamanho do projeto em questão. Já a utilização de agentes na arquitetura proposta permite que o sistema possa se adaptar a mudanças sofridas ao longo do desenvolvimento do projeto, raciocinar sobre o histórico de projetos e aprender com ele.

## 1.1.

### Caracterização do Problema

De acordo com (Joslin e Poole 2005), estimativas de duração de tarefas e recursos necessários na engenharia de software são muitas vezes imprecisas,

enquanto o gerenciamento efetivo de um projeto de software deve ser muito dinâmico. Já em (Lee e Miller 2004) é dito que o gerenciamento de múltiplos projetos é crucial, visto que um projeto pode afetar a data de término de outros projetos e determinar o sucesso ou fracasso dos mesmos, devido à existência de recursos compartilhados. Além dos vários problemas envolvidos no gerenciamento de projetos, vale destacar que os mesmos possuem tamanhos diferentes. Projetos de pequeno porte não demandam tantas atividades quanto os de médio e grande porte.

Um projeto normalmente sofre perturbações externas, sejam elas voluntárias ou não. Com isso, as incertezas acerca de um projeto devem ser constantemente monitoradas e o planejamento ajustado durante todo o seu ciclo de vida (Joslin e Poole 2005). O termo “reconfiguração dinâmica” no contexto pode ser definido por qualquer alteração feita sobre um planejamento prévio de atividades e recursos associados a um projeto. Em outras palavras, a reconfiguração dinâmica em projetos de software lida com eventos, ações, elementos afetados e consequências dos ajustes no processo de desenvolvimento de sistemas durante sua fase de execução (Callegari e Bastos 2008). Tal termo também tem sido utilizado em outras áreas, como computação autônoma (Horn 2001) e sistemas distribuídos (Coulouris, Dollimore e Kinberg 2002). Embora a definição atual possua pequenas diferenças das adotadas em tais áreas, o conceito principal do termo consiste na ocorrência parcial ou total de modificações durante a fase de execução. Atividades dos mais variados tipos devem ser associadas a recursos com características particulares a fim de atingir os objetivos relacionados a custo e prazo. O gerenciamento de projetos deve refletir dinamismo porque os projetos são intrinsecamente dinâmicos. Tal dinamismo dificulta a adaptação do gerenciamento ao longo do desenvolvimento, uma vez que em geral a identificação e tratamento dos problemas acontecem tardiamente. Quanto mais tarde os problemas são sanados, mais difícil a recuperação do andamento normal do projeto e maior a chance de comprometimento no custo e prazo do mesmo. Devemos ainda considerar que decisões corretas individualmente podem não ter a mesma representatividade coletivamente.

Dessa forma, existem muitos problemas no gerenciamento em um cenário de multi-projetos que devem ser considerados e sua resolução é fundamental para garantir o sucesso do desenvolvimento do projeto, como por exemplo: predição de prazos, alocação adequada de recursos, realocação de recursos,

replanejamento de atividades e uso de técnicas adequadas de acordo com o tamanho do projeto.

Apesar de cada um dos problemas citados serem individuais, é difícil tratá-los de forma isolada. Há a necessidade de se buscar uma solução de forma coletiva, uma vez que cada um dos problemas pode afetar os outros (e.g. recursos compartilhados em atividades de projetos distintos).

## 1.2.

### Limitação dos Trabalhos Relacionados

Muitos problemas envolvidos no gerenciamento de projetos vêm sendo largamente explorados na literatura. Soluções envolvendo heurísticas e outras metodologias são apontadas para solução dos mesmos (e.g. (Alcaraz e Maroto 2001)). Essas abordagens são complementares à solução aqui proposta, podendo ser integradas ao mecanismo de raciocínio dos agentes.

Em (Callegari e Bastos 2008), é feita uma revisão sistemática acerca de reconfiguração dinâmica em projetos de software. Com base nesta revisão, o artigo conclui que o problema multifacetado está longe de ser solucionado, visto que mesmo nos trabalhos mais recentes não é apresentada uma solução que trate simultaneamente todas as vertentes, ou mesmo várias delas. O que se percebe é o aprofundamento em muitos deles em cada caso isolado, mas não considerando o gerenciamento como um todo. Uma área clara para trabalhos futuros é a integração de soluções para os diversos problemas levantados no gerenciamento de projetos. Existem ainda problemas abordados que não tiveram solução proposta em nenhum dos artigos considerados, como por exemplo a alocação de um mesmo grupo de pessoas a atividades relacionadas com o objetivo de reduzir o tempo de início de novas atividades. Outro ponto destacado por (Callegari e Bastos 2008) é a necessidade de maior estudo para integração do planejamento de soluções com as atividades comuns de uma organização.

## 1.3.

### Solução Proposta

Diante do contexto exposto, é proposta uma linha de produtos de software dotada de agentes para gerenciamento de projetos de software que seguem o processo de desenvolvimento Rational Unified Process (RUP). Tal ferramenta tem o intuito de auxiliar os gerentes de projetos de software na tomada de decisões acerca do planejamento de atividades e dos recursos envolvidos. Dentre as diversas vantagens no uso de agentes, podemos citar o uso de

adaptação. Agentes são capazes de realizar modificações em algum grau no seu comportamento em resposta a mudanças no ambiente e em outros agentes. Além disso, um SMA utiliza o conceito de papéis como uma abstração de primeira classe. Tal abstração juntamente com a de organizações são intrínsecas ao problema apresentado neste trabalho. Como o contexto geral engloba diversas variáveis que afetam os projetos, o comportamento pró-ativo dos agentes traz benefícios à medida que os problemas podem ser detectados e tratados com técnicas de inteligência artificial (e.g. aprendizagem e raciocínio baseado em casos) (Wooldridge 2001). O uso de LPSs, como citado anteriormente, foi escolhido para tratar a variedade de porte (pequeno, médio e grande) dos projetos, uma vez que a idéia central da mesma é desenvolver uma infraestrutura reusável que provê suporte ao desenvolvimento de software de uma família de produtos (Nunes et al. 2009a).

Uma vez que os trabalhos relacionados apresentam soluções para somente alguns dos problemas levantados, a LPS proposta visa tratar um maior número de questões do gerenciamento de projetos. A solução proposta considera o cenário de multi-projetos, realocação de recursos, identificação de causas de atraso e predição de prazo para atendimento dos projetos. Para tal, os trabalhos analisados em (Callegari e Bastos 2008) servirão como base para tratar cada problema isoladamente. O uso de heurísticas ou de técnicas de pesquisa operacional será considerado para auxiliar no tratamento das diversas variáveis envolvidas no gerenciamento dos projetos, embora não haja pretensão de obter a “melhor solução possível” para cada um dos problemas. Algumas características dos projetos, como complexidade e prioridade, podem ser levadas em consideração para enriquecer a solução. Com relação a recursos, podemos ressaltar: quantidade, disponibilidade (e.g. férias e folga) e capacidade (rendimento por papel).

#### **1.4. Estrutura da Dissertação**

Esta dissertação está estruturada como segue.

A Seção 2 faz uma revisão teórica dos temas envolvidos neste trabalho: Gerenciamento de Projetos, Reconfiguração Dinâmica, Rational Unified Process, Sistemas Multi-agentes, Linhas de Produto de Software e Linhas de Produto de Sistemas Multi-agentes. Ao apresentar os temas citados, detalhes relevantes ao trabalho são abordados.

A Seção 3 traz detalhes da solução proposta. A linha de produto de sistemas de gerenciamento de projetos de software baseados em agentes é apresentada através de uma visão geral e de detalhes sobre seus agentes e diagramas.

A Seção 4 detalha o estudo de caso piloto feito para avaliação da solução proposta. Tal estudo de caso consiste em uma derivação da linha de produto para tratar projetos de médio porte.

Por fim, a Seção 5 traz a conclusão sobre o trabalho, suas contribuições e trabalhos futuros.