

1 Introdução

No processo de desenvolvimento do software, atividades relacionadas ao Processo de Requisitos envolvem atividades de elicitación, modelagem, verificação e validação dos requisitos [Leite94]. Tais atividades, pela sua própria natureza, são mais intensivas em comunicação que as demais atividades em um processo de desenvolvimento de software.

No contexto do processo de requisitos, tomemos como exemplo especificamente as atividades de verificação e validação (V&V). O processo de validação envolve diferentes atores, como por exemplo o engenheiro de requisitos, representantes de clientes e usuários e o coordenador ou gerente do processo de desenvolvimento. Nas atividades de verificação, por outro lado, interagem diferentes atores: inspetores, secretário, coordenador, autor do documento. Cada um desses atores participa visando atingir seus próprios objetivos, e a colaboração entre eles é necessária para atingir a meta maior associada a cada processo. Na verificação, a meta envolve obter respostas à pergunta "Estamos construindo o produto corretamente?". Na validação, a pergunta a ser respondida é "Estamos construindo o produto desejado pelos clientes e usuários?".

Dificuldades relacionadas às atividades de comunicação e coordenação de projetos são exacerbadas em ambientes distribuídos de desenvolvimento de software [Carmel99] [Paré99] [Bianchi02] [Damian03] [Herbsleb03]. Diferenças culturais implicam em dificuldades de compreensão da linguagem natural utilizada nos documentos de requisitos e diferenças em fusos horários podem dificultar a comunicação síncrona entre os interessados. Nesse cenário, praticamente inexitem os encontros informais entre desenvolvedores, que muitas vezes resultam no compartilhamento do conhecimento relacionado ao projeto ou mesmo em auxílio na resolução de problemas. Encontros virtuais ou troca síncrona de informações não substituem o relacionamento face a face, e desta forma a confiança entre os interessados e o espírito de grupo são afetados.

As restrições e dificuldades hoje encontradas em ambientes distribuídos de desenvolvimento têm levado muitas empresas a deslocar equipes para interagir diretamente com os clientes e usuários, evitando parte das dificuldades simplesmente pela eliminação da distribuição das atividades do processo de requisitos [Zowghi02] [Damian03a] [Lopes05].

A orientação a agentes tem sido apresentada como adequada para tratar problemas complexos ou de natureza intrinsecamente distribuída [Jennings00] [Message01] ou ainda sistemas nos quais atores desempenham diferentes papéis e interagem visando atingir diferentes objetivos [Cysneiros03]. O uso de agentes tem sido visto também na solução de problemas relacionados ao processo de desenvolvimento de software [Dellen96] [Gaeta02] [Grundy05] [Wongthongtham06].

Documentos de requisitos e outros artefatos manipulados no processo de requisitos costumam ser escritos em linguagem natural. A utilização de uma linguagem formal para registro dos requisitos traria a facilidade de uma verificação automática, mas dificultaria o processo de validação, pois clientes e usuários precisariam conhecer a linguagem formal utilizada. O uso da linguagem natural facilita a comunicação entre os participantes do processo, além de possibilitar que clientes e usuários validem o documento de requisitos sem necessitar de conhecimento extra. Técnicas de processamento da linguagem natural podem ser aplicadas a documentos de requisitos e outros artefatos, apoiando atividades relacionadas aos processos de verificação e validação de requisitos.

Propomos neste cenário a utilização do paradigma de agentes em apoio aos interessados¹ em atividades relacionadas ao Processo de Requisitos, tratando especificamente da verificação e validação dos requisitos. Nossa abordagem inclui também identificação de aspectos de dependência entre requisitos, e compartilhamento efetivo de informações relacionadas ao sistema em desenvolvimento. Utilizamos técnicas da área de Processamento da Linguagem Natural (PLN) para implementação dos serviços de apoio às atividades de V&V, disponibilizados através de agentes de software [Sayão05].

¹ denominamos de interessados todos aqueles que possuem algum interesse no sistema em desenvolvimento (clientes, usuários e profissionais de software e/ou hardware)

Nosso principal objetivo é explorar positivamente as possibilidades trazidas pela aplicação de técnicas de processamento da linguagem natural, implementadas como serviços de agentes pró-ativos de software, para apoio aos processos de verificação e validação de requisitos, no contexto do processo de definição de requisitos. Com isso visamos à obtenção de um documento de requisitos que atenda às necessidades dos usuários (validação) e aos requisitos de qualidade esperados (verificação), respeitando a distribuição de atividades entre os participantes do projeto.

1.1.

Desenvolvimento Distribuído de Software (DDS): desafios para o Processo de Requisitos

Diversos relatos envolvendo estudos e experimentos identificaram nas dificuldades de comunicação os maiores desafios enfrentados pelo pessoal envolvido nas tarefas e atividades associadas à etapa de requisitos. Carmel [Carmel99] considera que os impactos trazidos pela distribuição do desenvolvimento estão relacionados à distância, diferenças culturais e de fusos horários entre interessados. Essas dimensões são discutidas no contexto do Processo de Requisitos em [Zowghi02], para quem distância e fusos horários impactam diretamente na comunicação entre interessados, afetando as atividades de elicitação, negociação e priorização de requisitos. Dentre as diferenças culturais, aquelas relacionadas a normas de comportamento, linguagem e costumes podem levar à falta de confiança entre equipes e tendem a impactar negativamente atividades do processo de requisitos.

Em [Prikladnicki04] são relatados os principais resultados de um estudo envolvendo duas organizações e vinte e dois desenvolvedores utilizando DDS (Desenvolvimento Distribuído de Software); os participantes consideraram que o maior desafio no desenvolvimento distribuído é a Engenharia de Requisitos. A distância entre os participantes dificulta a convergência de idéias e impacta na estabilidade dos requisitos; dificuldades relacionadas à comunicação e falta de confiança entre equipes são menores quando há um padrão bem definido a ser seguido, tanto para o processo de desenvolvimento em si quanto para os artefatos gerados.

[Damian03a] apresenta resultados de um estudo de caso abrangendo quatro

sites situados em diferentes países, com a participação de vinte e quatro interessados. Dificuldades de comunicação face-a-face tornam a interação entre grupos dependente das características das ferramentas disponíveis, e interferem diretamente nas várias etapas do Processo de Requisitos. Sem um adequado compartilhamento da informação, a confiança entre equipes é atingida, os encontros virtuais necessários para atividades como priorização e negociação de requisitos tendem a não ser efetivos. A diversidade cultural, em especial diferenças lingüísticas, afeta a compreensão comum dos requisitos e a convergência entre diferentes interesses. *Delays* de tempo devidos aos diferentes fusos horários impactam nas atividades de priorização e negociação, se não devidamente gerenciados. Uma das principais contribuições desse trabalho é discutir e apresentar de forma esquemática as dimensões de comunicação, conhecimento, cultura e diferenças temporais no contexto do desenvolvimento distribuído, bem como identificar claramente as atividades afetadas.

Nossa proposta de trabalho está diretamente relacionada às dificuldades de comunicação entre interessados, e a aspectos do processo de requisitos que são afetados por esses problemas.

Comunicação é fator crítico de sucesso em projetos distribuídos: já em 1996 Gorton e Matwani [Gorton96] apresentaram resultados de um experimento que aponta como sendo de 22% o tempo utilizado em comunicação entre os membros de uma equipe. A maior parte desse tempo, cerca de 17%, foi utilizado em comunicação assíncrona, com uso de e-mail e de uma ferramenta de *groupware* desenvolvida especialmente para a empresa. Um outro estudo, feito por Cherry e Robillard [Cherry04], identificou que atividades de *cognitive synchronization* são responsáveis por aproximadamente 29% do tempo de desenvolvimento de um software. Por *cognitive synchronization* entende-se atividades de comunicação entre dois ou mais desenvolvedores, visando apenas confirmar que eles compartilham o mesmo conhecimento ou a mesma representação do objeto em questão. Outros estudos relacionados em [Cherry04] apontam para atividades de comunicação consumindo de 15 a 41% do tempo total do desenvolvimento.

A proposta de [Lopes05] para o Processo de Requisitos de certa forma corrobora o que foi apontado em [Cherry04]. Nessa proposta, uma equipe co-localizada aos usuários elabora o Documento de Requisitos (SRS ou Software Requirements Specification) e o repassa à equipe que executará o

desenvolvimento. Esta última equipe, após analisar os requisitos relacionados na especificação, identifica as possíveis fontes de problemas devidos por exemplo à ambigüidade ou falta de clareza na definição dos requisitos. Após a reescrita, o documento de requisitos é objeto de negociações entre as duas equipes até que a equipe de desenvolvimento tenha confiança no seu entendimento acerca do estabelecido no documento de requisitos.

Nossas observações junto a duas empresas que utilizam DDS e vários relatos mostram que é bastante comum a situação de uma equipe elaborando o documento de requisitos e posteriormente repassando esse documento para a equipe de desenvolvimento [Zowghi02] [Damian03a] [Prikladnicki03] [Audy04a]. Também encontramos a situação onde a equipe responsável pelo processo de requisitos se desloca até os clientes e usuários para a elicitação dos requisitos, retorna e trabalha este documento junto com a equipe de implementação, diminuindo desta forma os problemas decorrentes dos fatores culturais e lingüísticos para os demais participantes do desenvolvimento. Nestas situações, não há distribuição geográfica nas atividades do processo de requisitos. Acreditamos que isto se deve em parte à inexistência de ferramentas para dar suporte adequado à distribuição dos interessados e em parte para evitar os problemas de comunicação decorrentes da distância.

1.2.

Agentes nas Atividades de Verificação e Validação de Requisitos no Desenvolvimento Distribuído de Software

O paradigma de orientação a agentes para desenvolvimento de software tem sido tópico de trabalhos desde o início da década de 80 [Faltings00]. A literatura aponta para diversas e às vezes conflitantes definições para o termo *agentes*, mas a propriedade de *autonomia* é apontada como fundamental para muitos autores [Wooldridge99a]. O termo *autonomia* é utilizado para indicar que agentes podem agir sem necessitar da intervenção de usuários humanos ou outros sistemas. Aplicações utilizando agentes são encontradas em diferentes domínios, e recentemente no domínio da própria Engenharia de Software [Himmelspach03] [Dhavachelvan04] [Grundy05].

Wooldridge define agente como *um sistema de computação situado em algum ambiente e capaz de ações autônomas sobre este ambiente de forma a*

atingir os objetivos desejados [Wooldridge99a]. O agente monitora o ambiente através de sensores, e pode disparar ações que irão atuar sobre esse mesmo ambiente; a forma como isso ocorre aparenta não-determinismo. Um agente também é visto como um elemento interativo, adaptativo e autônomo que possui seu próprio ciclo de vida.

Em 2001 Jennings e Wooldridge [Jennings01] apontavam que a tecnologia de desenvolvimento orientada a agentes ainda estava num estágio inicial, mas que sua aceitação mostrava um futuro significativo na engenharia de software. Trabalhos posteriores confirmaram essa previsão, incluindo aplicações no próprio processo de desenvolvimento de software: [Gaeta02] utilizou agentes no contexto do desenvolvimento distribuído de software, com técnicas de *workflow* e de gerenciamento de documentos para apoiar a comunicação e o gerenciamento do projeto. Em atividades associadas à evolução de requisitos, [Chang01] mostra o uso de agentes móveis para controle da evolução de requisitos em ambientes distribuídos. [Wongthongtham06] utilizou agentes no contexto do desenvolvimento distribuído, com agentes manipulando um repositório controlado que armazenava uma ontologia de termos da própria engenharia de software.

Acreditamos que o uso de agentes é apropriado no contexto do Processo Distribuído de Requisitos [Chang01], porque o uso de agentes em sistemas de software propicia:

autonomia: agentes podem monitorar o ambiente, detectando modificações significativas ou ocorrência de eventos e comunicá-las aos envolvidos. No contexto em pauta, agentes podem monitorar mudanças em artefatos de requisitos, disparando atividades do processo de verificação e validação;

flexibilidade: agentes podem ser dinamicamente inseridos ou retirados do ambiente de execução, de forma transparente. Como agentes pessoais podem ser utilizados, e o conjunto de participantes no processo de desenvolvimento de um software evolui ao longo do processo de requisitos, o conjunto de agentes deve refletir essa evolução;

interatividade: agentes podem se comunicar através de mensagens; atividades de comunicação entre interessados são essenciais às atividades do gerenciamento de requisitos e agentes podem realizar parte dessas comunicações;

agregação do conhecimento distribuído: agentes podem colaborar e cooperar nas atividades que exigirem agregação de conhecimento para tomada de

decisão. O conhecimento relacionado ao sistema está distribuído entre interessados geograficamente dispersos, e agentes podem colaborar nessa agregação para a tomada de decisões;

acesso controlado à informação: definir restrições ao acesso de agentes à informação é tarefa facilmente realizável. No contexto de desenvolvimento de software, as restrições aos diferentes interessados com diferentes direitos de acesso às informações podem ser atendidas pelos agentes;

continuidade temporal: agentes podem estar ativos de forma contínua, monitorando os ambientes, detectando eventos e tomando decisões de forma independente, o que é essencial no caso do desenvolvimento *follow-the-sun*, quando os interessados geograficamente dispersos estarão ativos desempenhando diferentes tarefas durante as 24 horas do dia.

Nossa proposta visa repassar a agentes de software parte das tarefas e das comunicações que atualmente ocorrem entre agentes reais - os participantes do processo de requisitos - focando em atividades de verificação e validação. Visualizamos agentes de software como representantes dos diferentes atores nesses processos, propiciando a efetiva distribuição do trabalho no contexto do processo de requisitos e diminuindo os problemas de comunicação entre os interessados reais.

No processo de **verificação**, uma atividade possível é a inspeção do documento de requisitos. O processo de inspeção caracteriza-se pela utilização de uma técnica de leitura aplicável a um artefato, buscando a localização de defeitos ou inconsistências no mesmo, segundo um critério pré-estabelecido. Algumas das verificações estão relacionadas à omissão ou à duplicação de requisitos: o conjunto de requisitos é verificado em relação à completude (inexistência de omissões) e em relação à existência de duplicidade dos requisitos.

No processo de inspeção os diferentes atores envolvidos no processo de desenvolvimento vêem os artefatos sob diferentes pontos de vista. Tomemos como exemplo o documento de requisitos: ele será analisado através da visão de cada participante. O projetista vê esse documento considerando que pode utilizá-lo como base para a criação da arquitetura do software, considerando as funcionalidades e restrições descritas. O especialista em testes vê o documento como fonte primeira para a geração dos testes, que devem assegurar que o mesmo

atende às necessidades (funcionais e não funcionais) registradas. O engenheiro de requisitos vê o documento buscando identificar se ele atende aos requisitos de qualidade esperados.

A complexidade destas tarefas é amplificada se o número de requisitos a ser verificado ou inspecionado é da ordem de centenas ou mesmo milhares. O agrupamento de requisitos contribui não só para a organização do trabalho, através da geração de grupos menores de requisitos relacionados, mas também agiliza atividades próprias aos processos de verificação e validação, como por exemplo verificação da completude.

No processo de **validação**, engenheiro de requisitos e representantes do cliente e dos usuários avaliam a SRS e interagem com o objetivo de assegurar que os requisitos ali relacionados pelo engenheiro de requisitos correspondem ao esperado por usuários e clientes. Cada representante dos usuários traz a visão do grupo ao qual ele pertence (operacional, gerencial, decisório) e possui determinados interesses e necessidades em relação ao sistema a ser desenvolvido. Neste processo visualizamos o conjunto de requisitos dividido em agrupamentos que correspondem a diferentes características, facilitando para o cliente ou usuário o processo de identificar se as funcionalidades e restrições necessárias ao atendimento de seu trabalho estão refletidas no documento sendo avaliado.

Consideramos que as tarefas de verificação e validação podem ser parcialmente automatizadas por agentes de software. Nesse contexto, a cada ator corresponde um agente que age em seu nome, executando ações em acordo com a perspectiva de seu papel e seguindo o estabelecido em um plano pré-definido. Técnicas de processamento da linguagem natural podem ser aplicadas aos requisitos, possibilitando que parte das tarefas de V&V seja automaticamente realizada. Como exemplo, citamos a atividade de verificação de duplicidade em requisitos, que pode ser automatizada com a utilização de medidas estatísticas de similaridade.

Utilizamos a linguagem **i*** [Yu95] [Yu02] para modelar nossa proposta de uso de agentes para atividades de verificação e validação de requisitos. A Figura 1 apresenta o modelo de dependência estratégica dos processos de V&V de requisitos. Nesse modelo são mostradas as metas de cada agente e as dependências entre agentes na consecução desses objetivos ou metas. Os principais atores (cliente, desenvolvedor, engenheiro de requisitos, gerente do

projeto, inspetor) possuem relações de dependência por recursos, por *goals* (metas) ou por *softgoals*, que representam a intencionalidade dos atores.

Os principais atores envolvidos no processo de requisitos: clientes, usuários e profissionais de software, visualizados na Figura 1, ocupam diferentes posições. Profissionais de software ocupam posições que agregam responsabilidades por tarefas associadas ao gerente de projeto, ao engenheiro de requisitos e aos desenvolvedores. Clientes e usuários ocupam posições de representantes, agregando responsabilidades por tarefas de representação de diferentes perfis e interesses. O inspetor, papel que pode ser desempenhado por clientes, usuários e profissionais de software, possui responsabilidades específicas ao processo de verificação.

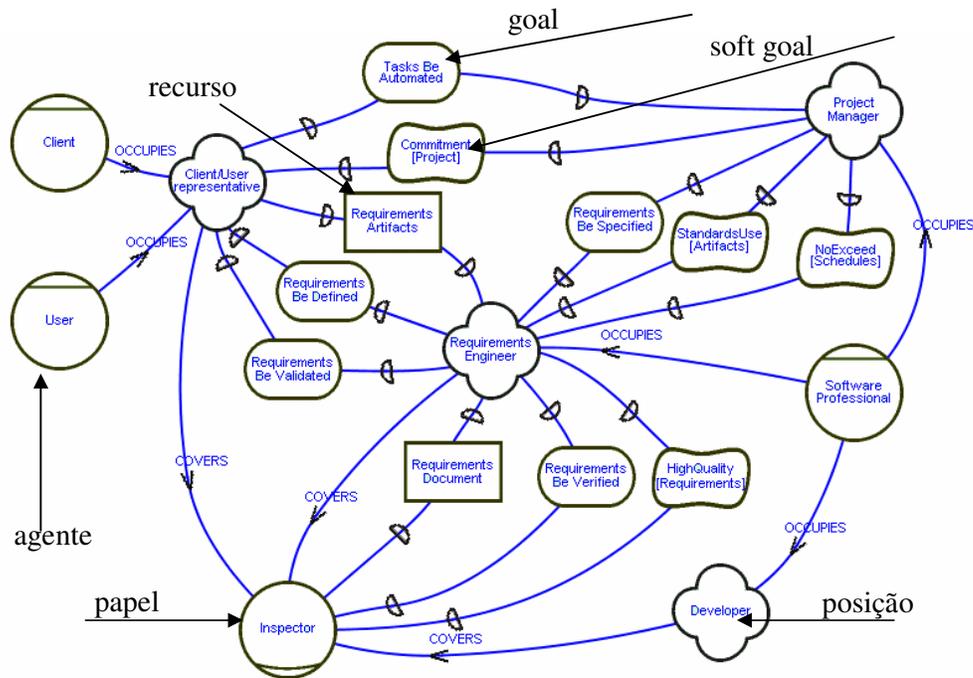


Figura 1 - Modelo de dependência estratégica para verificação e validação de requisitos

As relações entre atores reproduzem a rede de dependências existente entre eles, dependências estas relacionadas a metas, softgoals ou recursos. Numa dependência por metas um ator depende (*dependor*) de outro (*dependee*) para atingir objetivos concretos. Numa dependência por *softgoals* um ator depende do outro para satisfazer desejos ou intenções; numa dependência por recursos um ator depende de outro para obter um determinado recurso.

Podemos observar na Figura 1 que para atingir a meta de automação de

tarefas, representantes de clientes e usuários dependem do gerente do projeto. Este, por sua vez, depende do engenheiro de requisitos para que os requisitos sejam especificados atendendo aos padrões vigentes na organização e dentro do prazo previsto para a tarefa. O gerente de projeto depende também do comprometimento dos representantes de clientes e usuários para as atividades do processo de requisitos. O engenheiro de requisitos depende dos representantes de clientes e usuários que os requisitos sejam definidos e validados, e dos inspetores para que os requisitos sejam verificados e atendam a critérios de qualidade. Representantes de clientes e usuários dependem do engenheiro de requisitos para obter os artefatos que serão utilizados nas atividades de V&V, e o inspetor depende do engenheiro de requisitos para obter o documento de requisitos a ser verificado.

Em nossa proposta argumentamos que agentes de software são adequados para o atendimento de algumas atividades relacionadas à V&V. A implementação desta proposta envolve agentes e componentes inter-relacionados para atendimento das seguintes características:

agência: diferentes papéis a serem desempenhados pelos agentes, na representação dos interessados reais;

serviços de comunicação: para possibilitar a troca de informações entre usuários e agentes de software, e para notificação dos interessados na ocorrência de eventos;

serviços de gerenciamento: para possibilitar aos agentes o monitoramento de modificações no ambiente e eventos significativos nesse contexto;

persistência de dados: armazenamento de informações relacionadas a projetos e usuários e uma *baseline* para artefatos de requisitos [Leite95];

serviços de autenticação: para validação dos usuários aptos a utilizar o sistema;

geração de relatórios: para impressão de relatórios e de artefatos armazenados no sistema;

interface com usuários: para possibilitar fácil interação entre usuários e sistema;

Nossa argumentação e justificativa pela adequação do paradigma de agentes às atividades do processo de requisitos, num contexto de desenvolvimento distribuído, é detalhada no capítulo 4. Nossa argumentação considera que o

paradigma de agentes vem sendo apresentado, com sucesso, como alternativa para a modelagem e implementação de sistemas complexos e distribuídos, dado que (i) agentes autônomos podem atingir seus objetivos mesmo em ambientes dinâmicos, (ii) são capazes de interações em alto nível, e (iii) conseguem operar em estruturas organizacionais flexíveis [Jennings00].

1.3. Contribuições deste trabalho

A contribuição fundamental desta tese é a criação de um conjunto de procedimentos, técnicas e ferramentas para apoio às atividades de verificação e validação de requisitos. Parte destes processos fundamenta-se em estratégias de processamento automatizado da linguagem natural, dado que artefatos do processo de requisitos costumam ser modelados com uso da linguagem natural. Agentes de software implementam serviços que automatizam parcialmente os processos propostos.

De forma mais específica, várias contribuições são observadas a partir do trabalho desenvolvido:

- especificação de um **processo para a criação de uma taxonomia da aplicação**, baseada na extração de conceitos relevantes a partir de documentos de requisitos;
- utilização de **processo baseado nessa taxonomia para categorização de requisitos**, gerando agrupamento de requisitos semanticamente relacionados;
- proposta de um **processo para geração de visões dos requisitos**. Mapas textuais e gráficos são gerados e permitem uma melhor organização do trabalho nas atividades de V&V, além de possibilitar a visualização de interdependências entre requisitos, e a navegação através dos relacionamentos existentes entre a taxonomia e requisitos relacionados;
- desenvolvimento de um **processo para automatizar a construção e atualização do léxico da organização** a partir de documentos da organização e de artefatos gerados durante o processo de requisitos. Impactos de diferenças culturais entre participantes do DDS são minimizados quando o léxico inclui também expressões idiomáticas que podem levar a diferentes interpretações devido a fatores culturais;

- elaboração de **técnicas para apoio à detecção de discrepâncias, erros e omissões no documento de requisitos**. Aplicamos medidas de similaridade entre documentos para identificar duplicidade em requisitos, e elaboramos uma forma de identificar omissões em RNF's, tendo como base um dicionário de categorias (requisitos não funcionais) e expressões associadas. O documento de requisitos é avaliado em relação a esse dicionário, fornecendo um conjunto de relatórios e tabelas para subsidiar a análise de incompletude.

- **utilização de agentes na arquitetura da ferramenta de software** para suportar os procedimentos propostos. Na ferramenta visualizada, agentes interagem e fornecem serviços. Os serviços oferecidos incluem ferramentas para diversos tipos de tratamento da linguagem natural, cálculo de medidas de similaridade, notificação de interessados na ocorrência de modificações em requisitos. Os serviços oferecidos pelos agentes destinam-se a apoiar atividades de V&V do processo de requisitos. A ferramenta também é composta por agentes pessoais, destinados a representar humanos participantes do processo de desenvolvimento, visando diminuir sua carga de trabalho.

Até onde vai nosso conhecimento da literatura pesquisada, nenhum outro trabalho apresenta a utilização de agentes de software como apoio à geração de visões textuais ou gráficas de agrupamentos de requisitos, construção do léxico da aplicação ou mesmo atividades de verificação e validação de requisitos. Nesse sentido, as propostas apresentadas neste trabalho constituem uma nova abordagem, mostrando a adequação do uso de agentes como entidades autônomas, colaborando e apoiando atividades de verificação e validação de requisitos.

A adequação do paradigma de agentes para modelagem das funcionalidades providas neste trabalho é demonstrada através de um protótipo. O protótipo criado possibilita a oferta dessas funcionalidades em uma intranet ou na web. Isto apóia o reuso de funcionalidades disponibilizadas via serviços web e agiliza a construção de soluções para problemas mais complexos através da composição de serviços. A notificação automática dos interessados na ocorrência de determinados eventos pelo agente de comunicação poderá ainda levar à diminuição dos problemas de coordenação e comunicação entre participantes do processo de requisitos em ambientes distribuídos de desenvolvimento.

Uma das linhas de trabalho que aplica técnicas de PLN a artefatos de requisitos utiliza um sub-conjunto controlado da linguagem natural com o

objetivo de reduzir a ambigüidade nos requisitos. Diferentemente dessa linha, ressaltamos que a abordagem proposta trabalha com artefatos de requisitos escritos na língua portuguesa, sem quaisquer limitações ou restrições a nível de estruturas gramaticais ou vocabulário utilizados. Nossa abordagem também permite a utilização de requisitos expressos de diferentes formas, como sentenças de requisitos, casos de uso ou estórias do usuário.

1.4. Visão geral deste documento

No capítulo 2 são apresentados os principais desafios para a Engenharia de Requisitos no contexto do Desenvolvimento Distribuído de Software, e suas implicações para atividades de validação e verificação de requisitos. Este capítulo também relaciona trabalhos desenvolvidos com técnicas de processamento da linguagem natural em requisitos de software, uso de agentes para apoio às atividades próprias do processo de desenvolvimento de software e técnicas de processamento de linguagem natural que utilizamos nesta tese. Nossa proposta está apresentada em dois capítulos: o capítulo 3 apresenta os procedimentos desenvolvidos com uso de técnicas de processamento da linguagem natural, seguindo uma perspectiva de processo. O capítulo 4 detalha uma arquitetura baseada em agentes que atende aos processos propostos e vai além, incorporando características necessárias ao atendimento de necessidades específicas do desenvolvimento distribuído de software. Os estudos de caso realizados são apresentados no capítulo 5; o capítulo 6 apresenta trabalhos relacionados, e no capítulo 7 são relacionadas as conclusões e trabalhos futuros.