

2. Elicitação de Requisitos

2.1. Engenharia de Requisitos

O Dicionário Aurélio (Aurélio, 1999) define a palavra requisito, quando tomado por substantivo masculino, como uma condição necessária para a obtenção de certo objetivo, ou para o preenchimento de certo fim; quesito. Esta definição carrega o significado do termo de forma genérica na língua portuguesa, que comparado com as definições técnicas posteriores veremos não serem conflitantes. O modelo de maturidade CMMI (Capability Maturity Model Integration) versão 1.2 (CMMI, 2006) define o termo requisito em seu glossário como:

(1) A condition or capability needed by a user to solve a problem or achieve an objective. (2) A condition or capability that must be met or possessed by a product or product component to satisfy a contract, standard, specification, or other formally imposed documents. (3) A documented representation of a condition or capability as in (1) or (2).

Wohlin & Aurum (2005), observando a definição anterior, enfocam ainda que requisitos não incluem somente necessidades dos usuários, mas também aquelas que surgem de padrões organizacionais, governamentais e industriais em geral. O que os autores expõem nesta afirmação é o fato de que a rede de influências que pode afetar a construção de um *software* é bem maior que os seus usuários. Para os autores, um requisito é uma coleção de necessidades que surgem do usuário e de várias outras partes interessadas (organização em geral, comunidade, órgãos governamentais e padrões industriais), todas as quais devem ser satisfeitas.

Já Hull et al. (2005) definem o termo requisito como uma declaração que identifica uma capacidade, característica física, ou fator de qualidade que ressalta uma necessidade de um produto ou processo para o qual uma solução será adotada. Esta definição traz um aspecto interessante, pois fala das necessidades não só de um produto, mas também de processo, além de mencionar que um requisito pode, também, ser um fator de qualidade.

Tendo em vista que o sucesso de um projeto de *software* será medido a partir da satisfação de seus usuários e partes interessadas no mesmo, faz-se necessário ter uma abordagem disciplinada para definir estas necessidades. Portanto, a engenharia de requisitos é a disciplina que procura sistematizar a definição de requisitos (Leite 1994). O termo engenharia de requisitos refere-se a todas as atividades do ciclo de vida relacionadas a requisitos, que primariamente incluem a coleta, documentação e gerenciamento de requisitos (Wohlin e Aurum 2005). Definição semelhante também aparece em Sommerville & Sawyer (1997), para os quais a engenharia de requisitos é um termo que foi inventado para cobrir todas as atividades envolvidas na descoberta, documentação e manutenção de um conjunto de requisitos de um sistema baseado em computador. Para os autores, a utilização do termo engenharia implica que técnicas sistemáticas e repetíveis devem ser utilizadas para assegurar que requisitos de sistema são completos, consistentes e relevantes.

Como apontado nas definições anteriores, várias atividades podem compor um processo de engenharia de requisitos. A razão para tal é que existem diversas propostas de processos para engenharia de requisitos. Wohlin e Aurum (2005) enumeram as atividades de elicitação de requisitos, interpretação e estruturação (análise e documentação), negociação, verificação e validação, gestão da mudança e rastreabilidade de requisitos. Já Leite (1994) divide o esforço de engenharia de requisitos em elicitação, modelagem, análise e gerência dos requisitos.

O modelo CMMI versão 1.2 possui duas áreas de processo que lidam com requisitos, são elas as áreas de Gerenciamento de Requisitos e de Desenvolvimento de Requisitos. O objetivo da área de processo de Gerenciamento de Requisitos é gerenciar os requisitos de produtos do projeto e dos componentes de produto e identificar inconsistências entre esses requisitos, os planos de projeto e artefatos gerados. Esta área possui apenas um objetivo específico com suas práticas específicas associadas:

1. SG 1 Manage Requirements

1.1. SP 1.1 Obtain an Understanding of Requirements

1.2. SP 1.2 Obtain Commitment to Requirements

1.3. SP 1.3 Manage Requirements Changes

1.4. SP 1.4 Maintain Bidirectional Traceability of Requirements

1.5. SP 1.5 Identify Inconsistencies Between Project Work and Requirements

Então, a área de Desenvolvimento de Requisitos é uma área de processo do nível 3. Seu objetivo é produzir e analisar requisitos de cliente, de produto, e de componentes de produto. O modelo divide esta área nos seguintes objetivos específicos com suas práticas específicas:

- *SG 1 Develop Customer Requirements*
 - *SP 1.1 Elicit Needs*
 - *SP 1.2 Develop the Customer Requirements*
- *SG 2 Develop Product Requirements*
 - *SP 2.1 Establish Product and Product Component Requirements*
 - *SP 2.2 Allocate Product Component Requirements*
 - *SP 2.3 Identify Interface Requirements*
- *SG 3 Analyze and Validate Requirements*
 - *SP 3.1 Establish Operational Concepts and Scenarios*
 - *SP 3.2 Establish a Definition of Required Functionality*
 - *SP 3.3 Analyze Requirements*
 - *SP 3.4 Analyze Requirements to Achieve Balance*
 - *SP 3.5 Validate Requirements*

Tanto no objetivo específico “*Develop Customer Requirements*” como nas duas definições anteriormente citadas para quais as atividades que compõem um processo de Engenharia de Requisitos, a atividade de elicitação de requisitos aparece em comum. Este é o foco desta dissertação, mais especificamente uma parte desta atividade na forma como é definida por Leite (1994), que é a identificação e seleção de fontes de informação.

2.2. Elicitação de Requisitos

O termo elicitar, de acordo com Leite (1994), pode ser definido como: definir, tornar explícito, obter o máximo de informação sobre o objeto em questão. Também no Dicionário Aurélio (Aurélio, 1999), encontra-se dentre outras as seguintes definições do termo: fazer sair; extrair uma resposta ou reação

de um informante, extrair enunciados ou julgamentos lingüísticos de um informante.

A elicitação de requisitos é o processo de buscar, descobrir, adquirir, elaborar requisitos para sistemas baseados em computador (Zowgui e Coulin, 2005). Para os autores ainda se aplica a seguinte observação sobre o termo elicitação: “It is generally understood that requirements are elicited rather than just captured or collected. This implies there are discovery, emergence, and development elements in the elicitation process.”.

Para Leite (1994), dentro da engenharia de requisitos, cabe à elicitação a tarefa de identificar os fatos que compõem os requisitos do sistema, de forma a prover o mais correto e completo entendimento do que é demandado daquele *software*.

A elicitação de requisitos é a primeira atividade no processo de engenharia de requisitos, na qual se busca entender quais são as necessidades do usuário que devem ser atendidas pelo *software* que será desenvolvido (Sommerville e Kotonya, 1998). Este é um ponto em que as propostas de processos de engenharia de requisitos estão de acordo. Mesmo nos mais recentes métodos ágeis, a elicitação de requisitos é primeira etapa a ser executada dentro do ciclo de entrega de uma versão do produto. Em Extreme Programming (conhecida pela sigla XP) (Extremeprogramming.org, 2008], por exemplo, a elicitação de requisitos é o primeiro passo no ciclo de desenvolvimento de uma versão. Neste caso, são utilizadas as *user stories*, ou semelhantes, que são o ponto de partida tanto para a coleta e o desenvolvimento dos requisitos de *software* (Tomayko, 2002).

A atividade de elicitação de requisitos pode ser feita de diversas formas e diversos fatores podem influenciar a mesma, notoriamente, questões relacionadas ao ambiente no qual a elicitação de requisitos é feita e a finalidade para qual o sistema se destina. Por esta razão, os problemas da elicitação de requisitos não podem ser resolvidos de uma maneira puramente tecnológica, porque o contexto social é bem mais crucial do que nas fases de programação, especificação e desenho da solução (Goguen, 1994). Id. (1994) é um dos trabalhos que mais explorou o tema e apresenta desde as técnicas mais conhecidas até as técnicas advindas da sociologia que, segundo o autor, poderiam ser utilizadas com sucesso na elicitação de requisitos.

De acordo com o autor, as técnicas mais comuns utilizadas em elicitación de requisitos são as de introspecção, entrevista e análise de protocolos. A técnica de introspecção se baseia em imaginar que tipo de sistema eu iria querer se eu estivesse executando esta tarefa, utilizando este equipamento, entre outros. Ou seja, imaginar que propriedades um sistema deveria possuir de forma a obter sucesso. As entrevistas podem ser divididas em entrevistas com questionários, entrevistas livres e grupos focais e de desenvolvimento de aplicações. As entrevistas com questionários são amplamente utilizadas e, segundo Goguen [1994], têm a vantagem de parecerem científicas porque as mesmas podem utilizar análise estatística. Nas entrevistas livres o entrevistador faz um pergunta e, então, permite ao sujeito entrevistado responder como ele deseje. O entrevistador pode se aprofundar em mais detalhes, mas não impõe os termos da entrevista. A técnica de grupos focais é um tipo de entrevista em grupo. Nesta técnica grupos são reunidos para discutir alguns tópicos de interesse do pesquisador. Já a técnica de análise de protocolos pede a um indivíduo que se envolva em uma tarefa, explicando ao mesmo tempo o seu processo de raciocínio.

Para Zowgui e Coulin (2005) o processo de elicitación envolve uma série de atividades que devem permitir comunicação, priorização, negociação, e colaboração com todas as partes interessadas. Ele também deve prover fortes fundações para a emergência, descoberta, e invenção de requisitos como parte de um processo altamente interativo. Os autores classificam as típicas atividades da elicitación de requisitos em cinco tipos fundamentais: compreensão do domínio da aplicação, identificação das fontes de requisitos, análise das partes interessadas, seleção das técnicas, abordagens e ferramentas a serem utilizadas, elicitación dos requisitos das partes interessadas e de outras fontes.

No *Rational Unified Process* (Oberg et al., 2000), a elicitación de requisitos ocorre dentro do fluxo de trabalho “Compreender Necessidades das partes interessadas”. Neste fluxo, o analista de sistemas (que neste caso desempenha o papel do engenheiro de requisitos) e as partes interessadas chave identificam partes interessadas adicionais, elicitam seus pedidos e determinam as principais necessidades e características através de entrevistas, *workshops*, *storyboards*, casos de uso de processos de negócio e outras técnicas. Um ou mais analistas de sistemas (no papel de engenheiros de requisitos) facilitam estas sessões. Estes

workshops de requisitos, na visão do RUP, estão entre as técnicas de eliciação de requisitos mais úteis.

Para Leite (1994), o processo de eliciação de requisitos é composto de três tarefas: identificação das fontes de informação, coleta de fatos e comunicação. A identificação de fontes de informação tem por objetivo definir o contexto no qual a engenharia de requisitos vai ocorrer. A coleta de fatos é feita sobre as fontes de informação que foram identificadas na fase anterior, e a comunicação tem o papel de permitir o diálogo tanto para entender como para confirmar as informações acerca dos fatos. Na próxima seção, detalharemos melhor o conceito de Universo de Informação e Fontes de Informação.

2.3. Universo de Informações e Fontes de Informação

A engenharia de requisitos não ocorre no vácuo, pois depende de um processo, o processo de engenharia de sistemas que define o contexto e objetivos de um artefato de *software*. Este contexto é chamado de Universo de Informações (UdI) (Leite e Freeman, 1991):

O UdI é o contexto no qual o software deverá ser desenvolvido e operado. O UdI inclui todas as fontes de informação e todas as pessoas relacionadas ao software, essas pessoas são também conhecidas como os atores deste universo. Ele é a realidade circunstanciada pelo conjunto de objetivos definidos pelos que demandam o software. (Leite, 1994)

Termos definidos como UdI são importantes para que saibamos o contexto em que estamos, os limites do nosso desenvolvimento e os objetivos do produto que estamos desenvolvendo (Id., 1994).

É importante notar que uma definição do UdI se obtém através de uma observação do mesmo. Na tarefa de modelar as expectativas de um usuário em um UdI, um analista de sistemas pode encontrar, e geralmente o faz, diferentes opiniões sobre o problema sendo tratado. Diferentes analistas de sistemas, quando modelando as expectativas do usuário no mesmo UdI, produzem diferentes modelos. O mesmo analista de sistemas, modelando o UdI, pode fazer o modelo de formas diferentes utilizando diferentes perspectivas (por exemplo, modelo de dados, modelo de processos) (Leite, 1989). O importante a ser notado é que,

independente do ponto de vista, o processo utilizado para construção do *software* ou natureza do *software*, o UdI sempre existirá (Leite, 1994).

O conceito de fonte de informação está intrinsecamente ligado ao conceito de UdI. O UdI contém todas as fontes de informação que iremos utilizar durante a fase de elicitação, mesmo que tenhamos que refazer este UdI durante esta fase (Id., 1994]. O conceito de fonte de informação se refere aos elementos que possam nos fornecer requisitos. É comum encontrarmos esses elementos com outros nomes, como fontes originais ou fontes de requisitos (Gotel e Finkelstein, 1994). Em Sommerville e Sawyer (1997), são definidas as fontes de requisitos como uma ligação para a informação na qual o requisito é baseado. Nesta dissertação, utilizaremos a expressão fonte de informação e a definiremos como um elemento que faz parte de um Universo de Informação, podendo possuir ou não relações com outras fontes deste e a qual pode fornecer informação para a definição de requisitos do sistema.

Fontes de informação (ou fontes de requisitos) não estão restritas somente a pessoas, também referidas como partes interessadas, usuários ou clientes. Fontes de informação podem incluir diferentes tipos de material de leitura, documentos como: livros, memorandos internos, especificações de diferentes tipos, literatura científica, notícias, descrições de produto, manuais ou qualquer documento relevante para um melhor entendimento do nosso foco de interesse (UdI). Locais onde situações ocorram também podem ser uma fonte de informação, como o planto fabril, o escritório, a sala de reuniões, o balcão de uma loja, a sala de notícias de um jornal, ou qualquer lugar onde ações são executadas (Leite et al., 2007).

Zowgui e Coulin (2005) fazem observação semelhante quanto à natureza das fontes, para os autores as partes interessadas representam as fontes mais óbvias de requisitos para o sistema. Usuários e indivíduos especialistas no problema são utilizados para suprir informação detalhadas sobre o problema e as necessidades do usuário. De forma complementar ao que foi apontado como possíveis fontes de informação, os autores apontam também que os processos e sistemas existentes representam outra fonte para elicitação de requisitos, principalmente quando o projeto envolve a substituição de um sistema legado.

Para o RUP (Rational Unified Process) (Oberg et al., 2000), as principais fontes de informação a serem levadas em consideração são as partes interessadas.

Este por sua vez, como citado anteriormente, preconiza a participação das partes interessadas em *workshops* de requisitos como exemplo de partes interessadas ele cita usuários, pessoal da área de *help-desk*, donos do negócio, testadores e outros que têm interesse no resultado do projeto.

É importante saber que o UdI sempre existe. O UdI independe do modelo que está sendo utilizado. Segundo Leite (1994), mesmo que o macro sistema não esteja bem definido, sempre se poderá e deverá estabelecer os limites da atuação do engenheiro de requisitos. Antes que seja abordada a identificação de fontes de informação nesta dissertação, serão analisados os efeitos da identificação incorreta das fontes de informação e por conseqüente da não definição adequada do UdI, que será o tema da próxima seção.

2.4. Problemas relacionados às fontes de informação na literatura de requisitos

Os problemas apontados na literatura à cerca de fontes de informação podem ser divididos em dois grupos: o registro de fontes de informação para fins de rastreabilidade e a identificação e seleção das fontes de informação. Primeiramente, focaremos nas referências a cerca do primeiro grupo.

Gotel e Finkelstein (1995) realizaram uma análise dos problemas relacionados à rastreabilidade e abordaram as principais questões relacionadas ao tema. Para estes, a rastreabilidade de requisitos se refere à habilidade de descrever e seguir a vida de um requisito tanto para frente, como para trás. Os autores dividiram a rastreabilidade de requisitos em dois tipos:

- Rastreabilidade pré-especificação de requisitos (*Pre-RS traceability*): refere-se aos aspectos de vida de um requisito anterior a sua inclusão em uma especificação de requisitos.
- Rastreabilidade de requisitos pós-especificação de requisitos: refere-se aos aspectos de vida de um requisito após a sua inclusão em uma especificação de requisitos.

O termo especificação de requisitos aqui se refere ao documento de requisitos. O documento de requisitos é usado para comunicar os requisitos de

sistema para clientes, usuários de sistema, gerentes e desenvolvedores de sistema (Sommerville e Sawyer, 1997). Dependendo da organização, o documento de requisitos pode ter diferentes nomes como “especificação funcional”, “definição de requisitos”, “especificação de requisitos”, entre outros (Id., 1997).

Se fizermos uma comparação entre a definição de Sommerville e Sawyer (1997) para fonte de requisitos e a rastreabilidade pré-especificação de requisitos de Gotel e Finkelstein (1994), chegaremos à conclusão que esta parte de seu trabalho lida com o mesmo conceito, e, segundo os autores os problemas, na rastreabilidade de requisitos de pré-especificação são os mais complicados, pois a rastreabilidade pós-especificação é relativamente atendida pelas ferramentas e técnicas existentes no mercado. Dentre outros pontos, apontam que esta rastreabilidade é percebida como opcional, de baixa prioridade e por conseqüente com poucos recursos alocados a esta atividade, sem uma alocação e gerenciamento das diferentes tarefas a serem executadas, que são: obter, documentar, organizar e manter as informações necessárias.

Os poucos esforços em manter essa rastreabilidade são individuais, localizados e *ad-hoc* (Id., 1994). Outra questão abordada pelos autores é que a informação (como, por exemplo, o conhecimento tácito) nem sempre pode ser obtida e a qualidade da mesma varia. Culturas de desenvolvimento orientadas a entrega de produtos podem desencorajar a busca de certas informações a respeito das fontes de requisitos. Ainda assim, precisamos lembrar que métodos ágeis como XP (Extremeprogramming.org, 2008) altamente focados na entrega do produto, possuem uma cultura de cliente alocado junto à equipe de desenvolvimento. Na verdade, esta é uma forma de ter sempre à sua disposição a fonte de requisitos do sistema que está sendo desenvolvido. Gotel e Finkelstein (1994) também concluem que a documentação da informação necessária não é garantia de sua rastreabilidade. A questão é que não há garantia de que esta informação estará atualizada sempre.

O registro da fonte de requisitos também é apontada por Sommerville e Sawyer (1997) como uma boa prática a ser utilizada em engenharia de Requisitos. Os autores afirmam que um fator de custo majoritário na análise e mudança de requisitos é o custo da consulta às fontes de requisitos, a qual pode ser reduzida encurtando o tempo a quem ou quais documentos encontrar, caso estes estejam disponíveis imediatamente. Outro fator apontado pelos autores tem a ver com a

razão da existência dos requisitos. Eles apontam como exemplo um requisito cuja fonte seja um padrão de segurança, ao olhar a fonte, sabemos que aquele requisito se relaciona com questões de segurança.

O segundo grupo de problemas em relação às fontes de informação se refere à identificação e seleção das mesmas. Como já mencionado anteriormente, as fontes de informação podem ser de diversos tipos, desde pessoas (usuários, clientes testadores, analistas de sistemas, donos de negócio, entre outros), passando por legislações, documentação, técnica, processos de negócio ou relevantes para o problema em questão. Antonelli e Oliveiros (2002) realizaram uma pesquisa com objetivo de identificar quais as principais fontes de requisitos utilizadas por desenvolvedores de *software* na Argentina. A pesquisa foi realizada com 38 pessoas com formação acadêmica e experiência na indústria e o perfil dos participantes era variado em termos de experiência e tempo de graduação. A média de anos de experiência era de cinco anos e mais de 50% dos participantes tinham cinco anos ou mais de experiência no desenvolvimento de *software*. Para descobrir qual o tipo de fontes de informação mais utilizado pelos desenvolvedores, os autores sintetizaram as seis categorias de fontes de informação para requisitos propostos por (Loucopoulos e Karakostas, 1995, apud Antonelli e Oliveros, 2002], são elas:

- Especialistas no domínio
- Literatura sobre o domínio
- *Software* existente relacionado ao domínio
- *Software* de outros domínios
- Padrões nacionais e internacionais
- Outras partes interessadas do sistema de informação que hospedam o sistema de informações.

Partindo destes seis tipos básicos de fontes, os outros utilizaram uma taxonomia com quatro tipos de fontes de requisitos:

- Pessoas
- Formulários

- Conhecimentos adquiridos de desenvolvimentos anteriores
- Produtos do mundo real

O termo produtos do mundo real aqui descreve produtos externos ao sistema de *software* que incidem sobre o sistema de informação. Os produtos que os autores incluíram foram: leis, regulamentos, tratados, normas internas, padrões em geral, informação institucional, publicidade e o termo outros para que se pudesse incluir outra categoria que não fosse prevista. Tendo em vista essas fontes, foi pedido aos participantes que qualificassem cada fonte de acordo com o nível de contato com a fonte:

- Usa regularmente
- Usa ocasionalmente
- Usa alguma vez
- Conhece a fonte
- Conhece pessoas que a usam

Os resultados mostraram que as pessoas individuais são as fontes favoritas. Elas são utilizadas freqüentemente, 60 % das vezes. Grupos de duas a três pessoas vêm em segundo lugar com 42 %. E grupos maiores são utilizados apenas por 0,5 %. A pesquisa de Antonelli e Oliveiros (2002) é uma evidência da importância do uso de indivíduos como fontes de informação. Na literatura, os indivíduos que participam do processo de engenharia de requisitos são em geral vistos como as partes interessadas. Partes interessadas são pessoas que têm um interesse no sistema ou são afetadas por ele de alguma forma pelo desenvolvimento e implantação do sistema e, portanto, devem ser consultadas durante a elicitação dos requisitos (Zowgui e Coulin, 2005).

Pouloudi (1997) trata sobre a importância da identificação correta das partes interessadas na elicitação dos conhecimentos para a construção de sistemas baseados em conhecimento. Um dos problemas para o qual a autora chama a atenção é a seleção dos especialistas para participação no processo de desenvolvimento do *software*. A autora afirma que embora a indisponibilidade dos especialistas seja freqüentemente citada como um problema na literatura,

assume-se que o engenheiro do conhecimento não irá enfrentar dificuldades na identificação das fontes de conhecimento apropriadas. Mas a autora ressalta a complexidade que a identificação dos especialistas em casos particularmente complexos, nos quais múltiplos especialistas precisam ser considerados, ou em que sistemas baseados em conhecimento que serão entregues ou implantados devam ser afetados ou afetarem múltiplas partes interessadas.

As duas principais razões que Id. (1997) alega para tal situação são as ambigüidades sobre quem os especialistas são e a influência do ambiente no qual o especialista está trabalhando. Segundo a autora pode haver ambigüidade sobre quem são os especialistas, pois especialistas podem ser percebidos de formas diferentes por pessoas diferentes, tal que diferentes especialistas são considerados relevantes por diferentes partes interessadas na atividade de solução de um problema. Quanto ao ambiente, a atividade de solução de um problema deve ser abertamente ou implicitamente influenciada pelo ambiente no qual o especialista está trabalhando (Poulymenakou, 1995 apud Pouloudi, 1997). Neste caso, a forma pela qual a *expertise* é usada deve ser contingente sobre uma ampla faixa de partes interessadas, as quais seriam então necessárias serem identificadas como parte do processo de aquisição do conhecimento (Pouloudi, 1997).

Spence e Bittner (2002) afirmam que a importância das partes interessadas é grande no processo de desenvolvimento, pois elas são as fontes primárias de requisitos (fonte de informação). Para os autores as partes interessadas e os representantes das partes interessadas são os donos do problema e são afetados pelas soluções propostas. Os autores afirmam ainda que os representantes das partes interessadas que atuarão como fontes primárias de requisitos devem estar diretamente envolvidos no projeto e ter uma compreensão clara do papel que se espera que os mesmos desempenhem. Como exemplos do efeito da omissão de partes interessadas citam a questão dos usuários do sistema: se os usuários não são afetados materialmente pelo resultado do sistema, eles não utilizarão e o sistema em si, será um fracasso.

Também o modelo CMMI (CMMI, 2006) aponta para a importância da identificação de partes interessadas. A subprática “Identificar e envolver as partes interessadas relevantes do processo como planejado” da prática genérica “Institucionalizar um Processo Gerenciado” tem como propósito estabelecer e manter o envolvimento esperado das partes interessadas durante a execução do

processo, como descrito em um plano apropriado para tal. Segundo o modelo, as partes interessadas devem ser envolvidas em atividades como planejamento, decisões, comprometerimentos, comunicações, coordenação, revisões, avaliações, definições de requisitos, soluções de problemas.

Em pesquisa realizada por Juristo et al. (2002) foram contatados mais de 150 profissionais de organizações européias pra prover uma visão geral da situação atual em relação a problemas em engenharia de requisitos sem enfatizar os dados estatísticos. Os profissionais liberais foram contatados com base no seu envolvimento com o processo de engenharia de requisitos. A maioria possuía uma responsabilidade de média à alta no processo de engenharia de requisitos em suas companhias do ponto de vista do desenvolvimento de *software*. Os autores verificaram as respostas de 11 organizações em sete países da Europa de diversos mercados.

Uma das questões que foram avaliadas pelos entrevistados foi em relação a fontes de requisitos. Foi perguntado aos participantes se havia problemas na identificação de usuários e/ou partes interessadas e, que impactos os padrões, certificações e produtos de *software* possuem sobre o processo e os produtos do Engenharia de Requisitos.

Os resultados indicaram que as organizações devem considerar uma multiplicidade de fontes de requisitos incluindo fontes internas como o departamento de *marketing*, gerentes de produto, pessoal de vendas, clientes e fontes relacionadas como usuários e apoio ao usuário. Segundo os autores da pesquisa, baseados nas respostas colhidas, também devem ser considerados como fontes de requisitos:

- Padrões: três quartos das organizações pesquisadas consideraram o impacto dos padrões de grande importante. Padrões são úteis principalmente para organizar várias partes do processo, porque eles provêm uma lista de coisas para lembrar, guias para realizar tarefas, documentação e assim por diante.
- Leis: organizações devem antecipar futuras mudanças em leis e adaptar seus produtos de acordo.

- Certificação: certificação tem menos impacto nos requisitos. Um terço dos respondentes pareceu preocupado com questões relacionadas à certificação (como por exemplo, ISO 9000).
- Pacotes de *Software*: os respondentes reportaram que a utilização de pacotes de *software* mudou o processo de requisitos, porque o foco mudou das necessidades que o desenvolvedor tinha que satisfazer para as necessidades que o pacote de *software* disponível deveria satisfazer.

Id. (2002) cita Hofmann e Lehner (2001) como referência de que a identificação de fontes de requisitos tem sido uma prática bem sucedida na indústria. Id. (2001) realizaram um estudo de campo com objetivo de prover uma visão mais integrada do processo de Engenharia de Requisitos. Os autores investigaram o conhecimento da equipe, recursos alocados e os processos de Engenharia de Requisitos e sua contribuição para o processo de requisitos. Dentre as práticas associadas ao sucesso dos projetos, está a identificação dos limites da aplicação e das principais práticas interessadas.

Os autores apontam que as equipes de Engenharia de Requisitos bem sucedidas têm como fator de sucesso a prática de validar sua compreensão do domínio da aplicação com todas as possíveis fontes de requisitos as quais eles identificam. Eles examinam, por exemplo, artefatos de sistema com código fonte de sistemas anteriores. Entretanto, Juristo et al. (2002) afirmam que a multiplicidade de fontes de requisitos aumenta a complexidade do processo de Engenharia de Requisitos. O gerenciamento de múltiplas fontes e documentos de informação conflitante pode se tornar oneroso.

Como ocorre em Gotel e Finkelstein (1995), apesar de serem muito utilizadas dentro do processo de elicitação, indivíduos não são as únicas fontes de informação que podem se utilizadas no processo de engenharia de requisitos. As fontes textuais, ou seja, material de leitura, também são muito comuns no processo de engenharia de requisitos. Três dos quatro itens supracitados (Padrões, Leis e Certificação) provavelmente são elicitados pelas equipes de requisitos na forma de texto. Segundo Sawyer et al. (2005), o fato de o analista de requisitos ter que lidar com diversas entradas do processo de requisitos é um problema ainda mais árduo do que ter de processar os produtos do processo. Existe tipicamente uma ampla variedade de documentos que o analista precisa assimilar.

Transcrições de entrevistas ou relatórios de outros exercícios de elicitación têm de ser analisados. Os autores citam ainda que códigos de práticas corporativas, padrões e legislações devem ser analisados para determinar as informações contextuais sobre as tarefas dos usuários e seu ambiente.

Em função da dificuldade existente para elicitatar fonte textual é comum encontrarmos abordagens de uso de processamento de linguagem natural. Goldin e Berry (1997) propõem uma ferramenta chamada AbstFinder, cujo objetivo é auxiliar na identificação de abstrações, que pode ser executada em toda e qualquer informação que pode ser obtida do cliente e seus representantes, incluindo usuários. Os autores chamam essa massa de informações de informação cliente (*cliente information - CI*). Esta *CI* deve ser aquilo que os usuários acreditam ser a descrição completa do sistema e deve ser entrada como texto ASCII puro; Interações verbais (entrevistas, por exemplo) devem ser transcritas para texto. A saída do AbstFinder vem em duas partes: a primeira resumindo as abstrações identificadas e a segunda parte fornece uma descrição completa de cada abstração.

Outra abordagem baseada em processamento de linguagem natural pode ser encontrada em Sawyer et al. (2005). Neste trabalho os autores mostram como técnicas de *language engineering* baseada em *corpore linguistics* são capazes de proverem suporte para Engenharia de Requisitos em sua fase inicial de uma forma que este é tolerante tanto com o volume e a qualidade do texto sendo analisado. Os autores construíram um conjunto de ferramentas de *software*, que através das técnicas citadas anteriormente, tem por objetivo produzir um conhecimento superficial sobre o texto sendo analisado. Um analista humano (engenheiro de requisitos) pode então utilizar este conhecimento superficial para ajudá-lo a construir um conhecimento profundo que ele necessita adquirir como primeiro passo na formulação dos requisitos. Esse conjunto de ferramentas gera três saídas distintas: a lista de frequências (que contém a frequência de ocorrência de cada termo que aparece no documento), as afinidades léxicas presentes no documento (representam o arranjo das palavras dentro do texto, em termos de requisitos o arranjo freqüente de duas ou mais palavras sugere que elas representam um termo significativo) e uma lista de ocorrência de palavras no contexto (esta lista permite visualizar a palavra no contexto do documento fonte em que ela ocorre).

É interessante notar que, em ambos os casos, o sucesso no uso desse tipo ferramenta está relacionado a dois fatores: o grau de acerto dos algoritmos usados

e as entradas que serão utilizadas. Ou seja, mesmo com a existência da ferramenta, é necessário algum processo de seleção das fontes textuais de requisitos que seriam utilizadas. Ao utilizarmos transcrições de entrevistas, por exemplo, teríamos que garantir que a pessoa certa foi entrevistada, para que as informações fornecidas pela mesma sejam fidedignas em relação aos fatos a cerca do problema em questão. Se a pessoa incorreta foi selecionada para a entrevista, as abstrações identificadas a partir de suas informações levariam à formulação de requisitos incorretos.

Já no caso do uso de outros documentos que não entrevistas, como padrões, legislações procedimentos operacionais e outros ainda assim seria necessário realizar uma seleção dos mesmos. Pois todos estes documentos possuem atributos que são reconhecidos pelas partes interessada, mas não podem ser avaliados por ferramentas automatizadas. Padrões como CMMI (CMMI 2006) possuem de forma geral diferente versões que podem estar relacionadas ao ano de lançamento da versão ou customizações do padrão para fins específicos. Para que a versão correta do padrão seja escolhida ela deve passar por um processo de seleção por um ou mais especialistas no domínio, ou até mesmo por discussões envolvendo partes interessadas. A questão principal aqui é que uso destas ferramentas não remove a necessidade de aplicação de técnicas específicas de seleção das fontes a serem usadas.

Outro trabalho que se utiliza do processamento de linguagem natural é descrito em Stone e Sawyer (2006). Os autores, utilizando uma técnica de espaço vetorial chamada LSA (*Latent Semantic Analysis*), construíram uma ferramenta para retrospectivamente identificar rastros de pré-especificação de requisitos a partir dos requisitos para suas respectivas fontes. Para os autores, os requisitos freqüentemente incorporam conhecimentos que o analista já possui ou descobriu a partir da análise do domínio do problema. Esta seria útil na visão dos autores em duas situações, a primeira na validação de requisitos (Gotel, Finkelstein 1994), pois como já citado é comum termos que reconstruir a compreensão de como os requisitos foram produzidos, a segunda em situações como evolução de sistema ou tratamento dos pedidos de mudanças nos requisitos, em que a procedência dos mesmos precisa ser compreendida. O autor também tem como hipótese que onde a procedência não pode ser estabelecida entre requisitos e suas fontes haveria uma indicação do uso de informação tácita durante o processo de requisitos.

Outra possibilidade de fonte de requisitos são os locais em que situações ocorram também como a planta fabril, o escritório, a sala de reunião, o balcão de uma loja, a sala de imprensa de um jornal ou qualquer lugar onde ações são realizadas (Leite et al., 2007). Goguen et Linde (1993) fazem menção à importância do ambiente natural no estudo de temas relacionados à ordem social. Os autores citam que para compreender a ordem social como uma realização de participantes, nós devemos estudá-la em seu ambiente natural. Id. (1993) afirma que é importante levar em consideração a quem pertence a ordem social assumida durante a elicitação de requisitos, na qual pessoas de duas ou mais comunidades possivelmente muito diferentes tentam construir uma compreensão que seja exequível para ambos.

Damian e Zowgui (2002), os autores baseados em evidência empírica, construíram um modelo de como a comunicação remota e a gestão de conhecimento, diversidade cultural e diferenças de fuso impactam de forma negativa sobre a coleta, negociação e especificação de requisitos. Usaram como estudo de caso um projeto que foi realizado por uma empresa multinacional de desenvolvimento de sistemas. A empresa possuía sede nos Estados Unidos e possuía equipes globais desenvolvendo aplicação de *software*. Neste estudo de caso, foi investigado um projeto no qual a coleta, o planejamento estratégico e a negociação de requisitos, o desenvolvimento, teste e integração ocorriam em estruturas distribuídas. Segundo os autores os resultados de sua pesquisa indicam que existem impactos significantes na colaboração entre grupos funcionalmente e geograficamente distribuídos, envolvidos na negociação dos requisitos em um mercado consumidor diverso. Dentre os fatores que influenciam, está a questão da diversidade cultural do cliente dos negócios, que é apontada em Goguen e Linde (1993) quando estes abordam as questões de ordem social.

Tendo em vista os problemas levantados anteriormente, uma série de propostas têm sido desenvolvidas com o objetivo de lidar com os problemas relativos a fontes de informação (fontes de requisitos). As preocupações destas estão relacionadas à identificação, seleção e registro das fontes de informação utilizadas no processo de engenharia de requisitos. Na próxima seção, faremos uma análise destas técnicas baseada na literatura publicada sobre o assunto.

2.5. Técnicas para Seleção e Registro de Fontes de Informação

Nesta seção detalharemos técnicas existentes na literatura que lidam direta ou indiretamente com o conceito de fonte de informação (fonte de requisitos). Iniciaremos pelas questões relacionadas ao registro para fins de rastreabilidade (Sommerville e Sawyer, 1997; Gotel e Finkelstein, 1994), passaremos pelos aspectos de identificação de partes interessadas com Pouloudi (1999) e Sharp et al (1999). Tendo em vista o volume muito grande de abordagens relacionadas a partes interessadas, não seria possível abordar todas as referências, ou propostas a respeito do tema. Pouloudi (1999) foi escolhida por fazer uma revisão também dos aspectos gerenciais do uso de partes interessadas, além de apontar diversos trabalhos na área das partes interessadas em sistemas de informação. Já Sharp et al. (1999) foi escolhido por ser citado na estratégia que será apresentada no capítulo 3. Faremos uma avaliação das fontes de informação utilizadas em Extreme Programming (extremeprogramming.org, 2008). Por último, a proposta de uso de pontos de vista, que estão diretamente relacionados a fontes de requisitos, de Kotonya e Sommerville (1998). Esta foi escolhida em função de ser mais recente se comparada a outras propostas já existentes sobre, e por referenciar uma ampla gama de técnicas que usam pontos de vista.

2.5.1. Prática para registro de Fontes de Informação de Sommerville e Sawyer (1997)

A primeira questão que iremos abordar nesta seção são as práticas recomendadas ou técnicas para registro de fontes de informação quanto à questão da rastreabilidade. Sommerville e Sawyer (1997) recomendam que sempre que seja utilizado um formulário de requisitos, haja um campo que se possa anotar a fonte dos requisitos. Os autores também recomendam que a fonte seja anotada na hora da coleta dos requisitos para evitar que o registro da fonte seja perdido ou, devido à falha humana, seja esquecido. As fontes dos requisitos tanto podem ser anotadas individualmente para cada requisito, como podem ser anotadas para um grupo de requisitos. Segundo os autores existem vantagens e desvantagens em se adotar uma ou outra estratégia:

- Anotação individual: a princípio a rastreabilidade da anotação individual permite que se vá direto a sua fonte quando há a necessidade de se modificar o requisito. Entretanto, é fato que um requisito não prove de uma única fonte e sim da interpretação de um conjunto de fontes de informação. E é necessário observar que indivíduos mudam de emprego, portanto uma fonte não deve ser registrada como o nome de uma pessoa, pois eles podem não estar disponíveis no futuro.
- Anotação para grupos de requisitos: se a informação sobre uma fonte de requisitos for associada a um grupo de requisitos, é mais fácil lidar com as mudanças na fonte. Por exemplo, os requisitos estariam organizados por uma visão. Neste caso, a informação das fontes, freqüentemente consiste de vários documentos ou partes interessadas. Há uma maior probabilidade de que uma parte interessada esteja disponível no futuro.

Os autores aconselham que também para fontes de informação do tipo pessoas, é importante a questão de anotar o papel que o indivíduo desempenha na organização, pois caso este não esteja disponível no futuro seu substituto poderá ser consultado. Quanto a fontes como documentos, é recomendado que cópias dos mesmos sejam mantidas em um local conhecido e de tal forma que estejam disponíveis para consulta no futuro.

Ainda segundo Id. (1997) o custo para introdução destas práticas é bem baixo, pois envolve a definição e redefinição do seu formulário de requisitos com um campo a mais para o registro das fontes. A aplicação desta prática não incorre em grandes despesas a mais no processo de elicitação, em que se registra uma fonte de informação à medida que se coleta os requisitos. Esta seria uma abordagem mais simples baseada apenas no simples registro das fontes, mas, são possíveis abordagens mais sofisticadas e que vão além na questão do rastreabilidade.

2.5.2. Estruturas de Contribuição Gotel e Finkelstein (1994)

Mas para Gotel e Finkelstein (1994) a simples prática de registro das fontes dos requisitos não é suficiente. Os autores encontraram em sua pesquisa sobre os participantes no Processo de Engenharia de Requisitos que quando não era totalmente ausente, a informação era inadequadamente descrita e mantida. Os registros típicos eram apenas uma lista de nomes em um campo “autor/proprietário” de um documento. Estes registros rapidamente se tornavam desestruturados, ininteligíveis e inacessíveis para análise. Descobriu-se que não era incomum para os produtos finais da Engenharia de Requisitos perderem detalhes sobre quem originalmente gerou um requisito ou quem estava envolvido em seu refinamento.

Id. (1994) identificaram que a maior parte dos problemas em requisitos na rastreabilidade se refere à rastreabilidade pré-especificação de requisitos. Segundo os mesmos, a pré-rastreabilidade é necessária para produzir melhorias na qualidade, pois assuntos previamente fechados poderiam ser tornados explícitos, possíveis de serem reabertos ou re-trabalhados, ou seja, é necessária na validação e verificação dos requisitos. Esta também provê maior alavancagem econômica, em função de que, para usar e manter uma especificação de requisitos na prática, é frequentemente necessário reconstruir uma compreensão de como esta foi construída, o que é em geral sujeito a erro e custoso (Id., 1994).

Em um trabalho posterior, Gotel e Finkelstein (1995) propuseram o uso de um conceito chamado Estruturas de Contribuição como uma possível solução deste problema. No trabalho anterior, Gotel e Finkelstein (1994) haviam chegado à conclusão que a principal razão dos problemas em rastreabilidade de requisitos se devia à incapacidade de acessar e localizar as fontes humanas de requisitos, as informações relacionadas a requisitos, e o trabalho relacionado aos requisitos. Os autores então propunham uma abordagem para modelar o contexto social que deu origem aos requisitos de forma explícita e, portanto, rastreável, baseada na modelagem da estrutura de contribuição por baixo do artefato de requisito.

Basicamente a abordagem envolve a ligação entre artefatos de requisitos tangíveis (contribuições) aos detalhes dos agentes que contribuíram para sua produção (contribuintes) usando relações de contribuição. O termo “Estrutura de Contribuição” se refere a todas as relações de contribuição definidas para um artefato (Id., 1995). Existem três capacidades fundamentais nas quais agentes

podem contribuir para artefatos que juntos clareiam a noção de contribuição. Estes são:

- Mandante (*Principal*): aqueles que motivaram a produção do artefato cuja posição e/ou a crença é estabelecida por isto, portanto responsáveis pelos seus efeitos e conseqüências.
- Autor (*Author*): aqueles que escolheram a estrutura do artefato e seu conteúdo, portanto responsáveis por sua forma e semântica.
- Documentador (*Documentor*): aqueles que registraram ou transcreveram o conteúdo do artefato, então responsáveis pela sua aparência.

A abordagem também recomenda a utilização de atributos para especificar outros detalhes a respeito destas capacidades. O objetivo destes atributos é qualificar os participantes do processo dentro destas capacidades. A capacidade Mandante pode ser qualificada pelo uso de assinaturas, que é comum dentro do processo de desenvolvimento:

- Aprovado pelo Mandante.
- Faltando aprovação do Mandante.
- Não aprovado pelo Mandante.

Já a capacidade de autor poderia ser qualificada de acordo com as relações que o artefato possui com outros artefatos em questão:

- Se nenhuma relação existe então o *status* do autor é Criador (interseção entre o papel de Mandante e Autor, ou seja, aquele que exerce ambos os papéis).
- Se relações existem, o *status* autoral é determinado relativamente à ampla gama de funções de comunicações de cada relação e, posteriormente, por seu propósito comunicativo.

Em relação àqueles que registram a informação, estes possuem diversos tipos e graus de comprometimento em relação a esta. Isto pode ser refletido no

resultado final e na sua confiabilidade. A capacidade Documentador pode ser qualificada para refletir os humores da transcrição, adaptando os tipos de humor indicando se o mantenedor do registro está ou não:

- Certo de que o conteúdo é verdadeiro (de modo enfático).
- Acredita que o conteúdo é verdadeiro (de modo afirmativo).
- Indiferente a verdade do conteúdo (incerto ou apenas relatando).
- Desinformado sobre a verdade do conteúdo (modo indefinido ou interrogativo).

Como exemplo de uso dos papéis acima, os autores mostram a seguinte situação. Suponha que Olly forneceu o seguinte requisito “o sensor precisa ser sondado uma vez a cada 20 microssegundos”, e Dave tomou nota desta afirmativa. Neste caso Olly desempenha os papéis de *Principal* e *Author*, e Dave de *Documentor*. Dave é comprometido com sua aparência física, portanto ele pode lidar com qualquer questionamento a respeito de sua tipografia e pedidos de modificação. Perguntas como “Por que vinte segundos?” ou mudar propostas como “Por que não fazer uma vez a cada dez segundos?” precisam recorrer a Olly, pois é ele que está comprometido com o conteúdo e é dele a posição que seria desafiada por qualquer mudança.

Os autores também utilizam relações entre os artefatos, com o objetivo de definir o espaço dos mesmos. Estas relações possibilitam distinguir artefatos originais de cópias. São sugeridas três amplas categorias de relações que descrevem diferentes estruturas e possibilitam diferentes tipos de rastreabilidade de requisitos. São elas:

- Relações temporais: que descrevem a estrutura histórica do desenvolvimento e provêm meios de rastrear a história do requisito.
- Relações de Desenvolvimento: que descrevem a estrutura lógica do desenvolvimento e provem meios de rastrear o fluxo dos requisitos.
- Relações auxiliares: que descrevem formas adicionais nas quais as informações de desenvolvimento estão relacionadas e provêm estruturas suplementares e formas de rastreabilidade.

Os autores implementaram a sua técnica através de um protótipo de ferramenta na qual a rastreabilidade de requisitos pode ser estendida através de estruturas de contribuição. A ferramenta se dividia em outras duas ferramentas: a extensão de rastreabilidade e o gerente de contribuições. A ferramenta de extensão de rastreabilidade provê uma interface de hipertexto para o repositório de projeto. Já o gerenciador de contribuições será responsável pela modelagem e manutenção das estruturas de contribuição. Este possuía regras que usavam a informações capturadas pelas marcas de hipertexto para construir modelos, determinar a capacidade padrão, papéis sociais e comprometimentos, e para possibilitar novas deduções.

Segundo os autores, esta abordagem oferece um meio prático para lidar com a ausência de informações necessárias, informação suplementar documentada, e lidar com lado humano da mudança e da gerência dos requisitos. Isto se dá porque, segundo os autores, é possível identificar, é possível seletivamente identificar os agentes mais apropriados para proverem informações ou se envolverem no processo de mudança, tal fato auxiliaria os requisitos a se manterem modificáveis e manuteníveis.

Os autores citam como possíveis desvantagens:

- Resistência Organizacional à abordagem: esta se dá em função de que padrões claros de responsabilização têm aspectos negativos e positivos. Os autores apontam o trabalho de Nissembaun (1994) sobre responsabilização em computação como evidência de tal fato.
- Custos e benefícios para os profissionais de requisitos: os autores acreditam que esta é uma questão a ser avaliada, mas ainda assim crêem que o esforço extra necessário é mínimo e prático. Embora os mesmos no artigo não houvessem ainda realizado um estudo em escala industrial.
- Confiabilidade das pessoas: outro problema pode advir não só da dependência das pessoas em instanciar o formato de contribuição, mas também em sua habilidade para caracterizá-lo no esquema apropriado.

Os atores de um sistema não são as únicas partes interessadas presentes na construção de um sistema, mas com certeza são partes interessadas de grande

importância que devem ser levadas em consideração. Por exemplo, a detecção de conflitos de interesse entre atores tem uma grande importância na área de segurança. A razão é que uma grande fonte de vulnerabilidades em sistemas é a presença de conflitos entre requisitos. Neste contexto, conflitos de interesse são críticos no desenvolvimento de sistemas seguros em função de que atores confiáveis constantemente abusam de sua posição para ganhar vantagem pessoal (Giorgini et al., 2008). Numa situação destas, a não identificação (ou seja, exclusão) de uma parte interessada pode levar a sérios problemas de segurança como, exposição de informações de indivíduos e organizações.

2.5.3. Identificação de Partes Interessadas (Pouloudi, 1999)

Na literatura de sistemas de informação, a noção de parte interessada tem sido utilizada de muitas formas diferentes, que freqüentemente tendem a refletir primariamente uma perspectiva descritiva ou instrumental (Pouloudi, 1997). Em Id. (1997) é feita uma extensa comparação entre os conceitos gerenciais de partes interessadas e de sistemas de informação. Para a autora, que baseia suas conclusões em estudo anterior (Pouloudi, 1998, apud Pouloudi, 1999), a identificação interpretativa de partes interessadas deveria ser dinâmica, dependente de contexto e iterativa. O processo de análise de partes interessadas deveria ser independente da identificação, uma vez que as partes interessadas têm sua visão sobre quem são as partes interessadas. A autora propõe um conjunto de princípios do comportamento de partes interessadas que deveriam ser utilizadas para guiar a identificação e análise das mesmas em sistemas interorganizacionais em um dado contexto.

Princípios do comportamento de partes interessadas	Implicações para a identificação e análise de partes interessadas
1. O conjunto e o número de partes interessadas dependem do contexto e do tempo	<ul style="list-style-type: none"> • O mapa de partes interessadas deve refletir o contexto • O mapa de partes interessadas deveria ser revisto com o passar

	do tempo
2. As partes interessadas não podem ser vistas isoladas	<ul style="list-style-type: none"> • Considere como as partes interessadas estão ligadas
3. O papel de uma parte interessada pode mudar com o passar do tempo	<ul style="list-style-type: none"> • Adote uma perspectiva de longo prazo: estude como as percepções mudam
4. Partes interessadas podem ter múltiplos papéis	
5. Partes interessadas diferentes podem ter diferentes perspectivas e desejos	<ul style="list-style-type: none"> • Existem diferentes versões do mapa de partes interessadas a serem desenhadas
6. Os pontos de vista e desejos das partes interessadas podem mudar com o passar do tempo	<ul style="list-style-type: none"> • Estas diferentes versões do mapa deveriam ser revistas com o passar do tempo
7. As partes podem não estar aptas a servirem seus interesses ou a realizarem seus desejos	<ul style="list-style-type: none"> • É necessário considerar questões políticas (tais quais econômicas, técnicas ou outras)

Tabela 1 - Propostas para identificação e análise de partes interessadas [Pouloudi 1999].

A autora enfatiza que uma faixa maior de partes interessadas pode oferecer uma referência mais diversa e rica de um contexto interorganizacional, especialmente se o pesquisador foca em coletar uma variedade de percepções e também explorar as mudanças dessas percepções através do tempo. Para a autora tal abordagem trata dos diferentes aspectos do conceito de partes interessadas e mostra suas inter-relações sem estar presa a perspectiva gerencial, mas ela ressalta que, de fato, em primeira instância esta abordagem provê uma perspectiva descritiva das partes interessadas.

2.5.4. Identificação de Partes Interessadas segundo Sharp et al. (1999)

Sharp et al. (1999) propõem uma abordagem para identificar todas as partes interessadas relevantes para um sistema específico, que os autores acreditam ser independente de domínio. O ponto de partida da proposta é um conjunto de partes interessadas chamadas de partes interessadas *baseline*. A partir destas podem ser identificadas partes interessadas do tipo “fornecedoras” e partes interessadas do

tipo “cliente”: a primeira provê informações ou tarefas de suporte para o *baseline*, e a segunda processa ou inspeciona os produtos do *baseline*. Existem ainda outras partes interessadas as quais são chamadas de satélites que interagem com o *baseline* em uma variedade de formas (comunicação, leitura, e um conjunto de regras ou guias procurando por informação, entre outros). A abordagem foca nas relações entre as partes interessadas ao invés de relacionamentos entre partes interessadas e o sistema, pois estas são mais fáceis de seguir.

Nesta abordagem, foram identificados quatro grupos de partes interessadas do tipo *baseline*: usuários, desenvolvedores, legisladores e tomadores de decisão. Os usuários são as pessoas, grupos ou companhias que irão interagir com o *software* e controlá-lo diretamente, e os que irão utilizar algum produto do sistema (informações, resultados, etc.). Os desenvolvedores são partes interessadas cujo interesse difere dos usuários na especificação de requisitos. Os legisladores são organizações profissionais, agências governamentais, associações comerciais, representantes legais, executivos de segurança, auditores de garantia de qualidade, entre outros que podem produzir normas de procedimento que afetarão o desenvolvimento e/ou a operação do sistema. Os tomadores de decisão são as estruturas internas à organização responsáveis pelo desenvolvimento ou organização dos usuários que se relacionam com o sistema que está em construção.

No método proposto existem cinco passos básicos para identificar as partes interessadas:

- i. Identificar todos os papéis dentro dos grupos de parte interessadas do tipo *baseline*.
- ii. Identificar as partes interessadas fornecedoras para cada papel do *baseline*.
- iii. Identificar as partes interessadas do tipo cliente para cada papel do *baseline*.
- iv. Identificar as partes interessadas do tipo satélite para cada papel do *baseline*.
- v. Repetir o primeiro e o quarto passo para cada grupo de partes interessadas identificadas do passo ii ao iv.

Para cada grupo de partes interessadas, existem notas específicas a serem seguidas. Para os usuários, o ponto de partida para o *baseline* desse grupo são os papéis diretos de grupo de usuários, que podem ser identificados pela frequência de uso do sistema, experiência no uso de computador, objetivos esperados, posição na organização, grupos internos ou externos à organização dentre outros.

Para os desenvolvedores, os papéis que compõem o grupo de *baseline* podem incluir analistas, *designers*, programadores, testadores, representantes de garantia de qualidade, mantenedores, treinadores, gerentes de projeto e assim por diante. Para os legisladores, identificar os papéis e os aspectos legais que eles cobrem. Como, por exemplo, a operação do sistema, como legislação de proteção de dados, desenvolvimentos padrões de defesa. Para os tomadores de decisão um bom ponto de partida é o usuário responsável por comissionar o sistema. Identificar os tomadores de decisão pertencentes à organização de desenvolvimento (que podem ser os mesmos ou diferentes) que tem qualquer poder sobre a decisão de construir o sistema, ou sobre quaisquer processos, pessoas ou padrões identificados em pesquisas anteriores sobre partes interessadas.

Segundo os autores, os benefícios potenciais desta abordagem são que ela começa com uma parte interessada conhecida como centro e, para os mesmos, este aspecto seria capaz de capturar as parte interessadas importantes e, ao mesmo não incluir partes irrelevantes. A checagem de consistência é construída dentro do processo porque as partes interessadas supridoras e as partes interessadas cliente podem se sobreporem, se existe um produto do sistema para o qual não existe usuário, então alguma coisa está faltando de sua análise.

O objetivo dos autores com essa abordagem é descobrir uma rede de partes interessadas, em que cada parte interessada seria um nó, e cada aresta será um relacionamento entre duas partes interessadas. Se uma teoria de redes adequada pudesse ser identificada, poderia ser possível automaticamente calcular o peso da visão de uma parte interessada em relação ao sistema, baseado nas relações entre as partes interessadas identificadas através de pesquisa. Tal fato levaria ao potencial de construção de suporte baseado em ferramentas para identificar partes interessadas importantes e seu nível de influência.

2.5.5. Fontes de Informação em Extreme Programming

Uma das práticas chaves de métodos ágeis como o XP (extremeprogramming.org, 2008) é a colaboração entre clientes e engenheiros que formam times trabalhando juntos. Os processos ágeis permitem aos requisitos de cliente emergir dinamicamente sem penalidade para o projeto (Schwaber, 2002). Dois ciclos colaborativos de trabalho ocorrem em processos ágeis. As equipes de desenvolvimento frequentemente iteram novos incrementos de funcionalidade. Os clientes iteram e priorizam uma lista de funcionalidades baseadas em custo, prazo, qualidade e condições emergentes do negócio (Id., 2002). Inferimos então que o cliente, no processo XP, é a principal fonte de informação utilizada pelos desenvolvedores. Uma vez que este está sempre alocado à equipe de desenvolvimento.

Em XP os clientes que estão alocados ao projeto levam para a reunião os requisitos do sistema, que XP chama de *user stories*, escritos em cartões de papel, pois permitem facilidade de manipulação e armazenamento. Estes cartões, chamados de *story cards*, pela metodologia, descrevem os requisitos de forma sucinta (Endriss, 2003). Estes por sua vez são utilizados pelos desenvolvedores para estimar a quantidade de semanas para implementar uma *user story*.

Para Tomayko (2002) as *user stories* ou similares são apenas os pontos de início do processo de busca e desenvolvimento de requisitos nos métodos ágeis. Estes requisitos iniciais são simplesmente um lugar para começar. É esperado adicionar mais requisitos à medida que se sabe mais a respeito do produto. Estas *user stories* também podem ser vistas como fontes de informação. Podemos inferir que elas se relacionam com as fontes de informação do tipo cliente no sentido que o cliente tem uma relação de autoria com as mesmas.

2.5.6. Engenharia de Requisitos com Pontos de Vista (Kotonya e Sommerville, 1996)

Para Kotonya e Sommerville (1996), o processo de engenharia de requisitos envolve uma clara compreensão dos requisitos do sistema pretendido. Isto inclui os serviços requeridos pelo sistema, os usuários do sistema, seu ambiente, e

restrições associadas. Para os autores este processo envolve a captura, análise e resolução de muitas idéias, perspectivas e relações em diferentes níveis de detalhe. Os autores afirmam que para tal tarefa os métodos baseados em um raciocínio global parecem carecer de um *framework* expressivo para adequadamente distribuir a estrutura de conhecimento de requisitos. Para os autores, a solução de alguns desses problemas passa pela utilização de uma abordagem orientada a pontos de vista.

A noção de ponto de vista como forma de organizar estruturas e a atividade de requisitos está presente em outros trabalhos (por exemplo, Leite (1989)). Mas para os autores, estas abordagens consideram os pontos de vista como depósitos ou fontes de dados, processo de subsistema ou perspectivas internas. Os autores então propõem uma noção de ponto de vista que se baseia nas entidades que serão responsáveis pelo, ou poderão restringir, o desenvolvimento do sistema pretendido. Estas fontes de requisitos compreendem usuários finais, partes interessadas, sistemas que fazem interface com o sistema proposto e outras entidades no ambiente do sistema pretendido que pode ser afetado por sua operação. Cada fonte de requisitos tem uma relação com o sistema proposto baseada em suas necessidades de interação com o sistema (Id., 1996).

Baseados nesta definição de pontos de vista, os autores desenvolveram um método para engenharia de requisitos o qual chamaram de VORD (Viewpoint-Oriented Requirements Definition) cujo objetivo é dar suporte desde a descoberta dos requisitos até modelagem detalhada do sistema. Em função do artigo que foi utilizado como referência e também do foco desta seção só serão detalhados os três primeiros passos iterativos do método, pois estes é que estão focados na questão da identificação de fontes de requisitos. Os três passos foram sintetizados e estão descritos a seguir:

- i. Identificação do ponto de vista e sua estruturação: este passo tem por objetivo identificar pontos de vista relevantes no domínio do problema e estruturá-los. O ponto de partida para a identificação do ponto de vista é feito com as declarações abstratas das necessidades organizacionais e as classes abstratas dos pontos de vista. Estas classes abstratas de pontos de vistas são uma generalização do conceito de fontes autorizadas do sistema (*system authorities*). Estas fontes são pessoas ou documentos com um

interesse em ou conhecimento de especialista no domínio da aplicação. A figura abaixo mostra a hierarquia dessas classes:

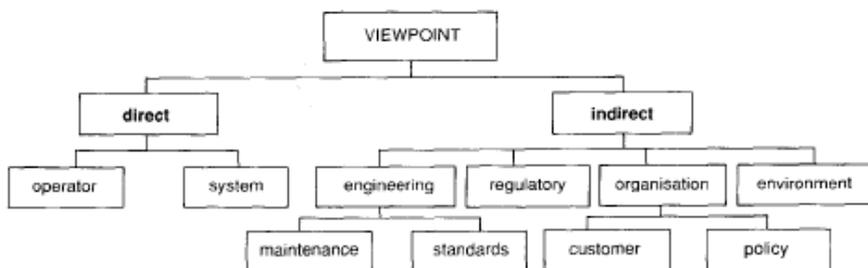


Figura 1- Classes abstratas de Pontos de Vista (Kotonya e Sommerville, 1996).

- ii. Documentação dos pontos de vista identificados no primeiro passo: a documentação do ponto de vista consiste em registrar o nome do ponto de vista, os requisitos, as restrições sobre seus requisitos e a fonte de seus requisitos. Requisitos de um ponto de vista compreendem um conjunto de serviços, requisitos de controle e o conjunto de requisitos não funcionais.
- iii. Especificação dos requisitos funcionais e não funcionais: os requisitos (funcionais e não funcionais) devem ser especificados em uma forma apropriada. O que os autores enfatizam neste ponto é que pode haver várias notações apropriadas. A notação mais adequada pode variar de linguagem natural (se a fonte de requisitos está preocupada com requisitos não-técnicos), equações (os autores citam o caso de a fonte de requisitos ser um físico, por exemplo), a modelos de sistema expressados em alguma linguagem formal ou notações estruturadas.

Os autores desenvolveram um modelo para a documentação dos pontos de vista que contém os seguintes elementos:

- Os requisitos associados com o ponto de vista; estes podem ser funcionais ou não funcionais.
- As fontes de requisitos associadas com o ponto de vista.
- A razão para os requisitos propostos.
- As restrições sobre os requisitos dos pontos de vista e suas fontes.

- Os eventos do ponto de vista: os eventos que os pontos de vista descrevem a interação entre o ponto de vista e o sistema pretendido, em termos de ventos do ponto de vista, respostas do sistema e exceções.

Ao final da aplicação destes passos, os requisitos são coletados em um repositório central para servirem de entrada para a fase de análise.

2.6. Comparação das técnicas pesquisadas na literatura

As técnicas abordadas nesta seção podem ser relacionadas em diversos aspectos. O conceito de registro de pontos de vista abordado por Kotonya e Sommerville (1999) é encontrado parcialmente esboçado na boa prática de registro de fontes de requisitos de Sommerville e Sawyer (1996). Para estes, quando agrupamos vários requisitos originados pela mesma fonte, estamos montando um ponto de vista. Já as abordagens de Gotel e Finkelstein (1995) provêm das mesmas necessidades de registro de fontes de informação abordadas por Sommerville e Sawyer (1996): melhor suporte à mudança nos requisitos e validação dos mesmos, uma vez que é comum ser necessário acesso à fonte original para reconstrução do contexto no qual o requisito foi formulado. A questão é que a abordagem de Gotel e Finkelstein (1995) vai além, guardando inclusive as informações a respeito do contexto em que surgiram os requisitos, e não só a fonte. A preocupação com as fontes de informação também está presente nas abordagens de métodos ágeis uma vez que esses preconizam que a fonte, no caso o cliente, deve estar sempre presente. Mas estas falham quanto à questão da rastreabilidade da fonte após o término do projeto.

Um fator é comum às abordagens de partes interessadas e de pontos de vista. Duas delas (Sharp et al., 1999; Kotonya e Sommerville 1999) se utilizam da mesma estratégia para identificar as partes interessadas ou fontes autorizadas do sistema. Tal fato se dá em parte porque os conceitos são bastante semelhantes. Com a diferença que as fontes autorizadas do sistema não estão restritas apenas a pessoas, elas podem ser documentos que forneçam informação na formulação de requisitos. A estratégia é a identificação a partir de um conjunto de papéis pré-definidos que a entidade desempenha no Universo de Informação. Sharp et al. (1999) utilizam o termo “Grupos de Partes Interessadas” os quais eles dividem em

linha base e satélite, que já foram abordados anteriormente. Já em Kotonya e Sommerville (1999), os autores chamam esses papéis de “Classes Abstratas de Pontas de Vista” cuja decomposição em categorias pode ser vista na figura 1.

Outro ponto em comum entre as abordagens acima e que também é a proposta de Pouloudi (1999) é a importância da exploração das relações entre as entidades de suas abordagens. Pouloudi (1999) ressalta que as partes interessadas não podem ser vistas em isolamento com a necessidade de desenvolverem ligações entre as partes interessadas. Sharp et al. (1999) vai mais além, inclusive averiguando a possibilidade da utilização de uma teoria de rede para descrever a importância das partes interessadas para o sistema.