Descrição: arte e técnica no descobrimento e comunicação de requisitos

22/01/2021

Autores: *Edna Pacheco Zanlorenci - GPS Robert Carlisle Burnett - PUC-PR*

Resumo

Requisitos representam a base de comunicação entre a demanda conceitual para o entendimento de problema, quer de desenvolvimento de produto ou de prestação de serviço e a efetiva estruturação de idéias para concretizá-los.

Descrever os requisitos é uma atividade extensa e intensivamente essencial, observados os cenários ambientais e os mais variados pontos de vista e intenções dos clientes e usuários.

Este artigo aborda as várias técnicas de descrição e método de estruturar o pensamento, aplicáveis na representação do domínio do conhecimento.

A título de ilustração dos conceitos, utilizam-se os requisitos para um sistema gestão de segurança e paisagismo da malha rodoviária na área de transportes.

O que e como descrever, fundamentalmente, constituem arte e técnica no descobrimento e comunicação dos requisitos para estruturação do problema e a sua representação.

Palavras-chave: ambiente, definição, descrição, designação, domínio do conhecimento, especificação, problema, requisito.

1. Introdução

Organizar, estruturar e fazer descrições são a atividade central de desenvolvimento de software. Um método de desenvolvimento de software estipula o que as descrições fazem para resolver o problema, em que ordem, por qual operação e em qual linguagem.

A satisfação dos requisitos demandados pelo cliente é a finalidade do desenvolvimento de software e um problema constante na Engenharia de Requisitos é a validação dos requisitos documentados, para a construção de produtos ou serviços de software. Isto implica em agregar esforços no sentido de

aprimorar o processo de descoberta do conhecimento, na descrição e validação das informações.

Dentro deste contexto, este trabalho apresenta-se como uma tentativa de buscar na origem do problema de representação do conhecimento os fundamentos conceituais que delineiam ações no processo descritivo, para a garantia de qualidade do produto resultante da extração de requisitos. O trabalho foi organizado em partes, assim distribuídas:

A parte 2 apresenta a contextualização do problema de desenvolvimento de software e a conseqüente habilidade do desenvolvedor de software no uso de técnicas na captação das informações e na representação do conhecimento. Também de igual importância a participação do cliente como agente da informação. A parte 3 apresenta a descrição, como arte e técnica na atividade essencial de qualquer processo de descoberta do conhecimento, como a linguagem para representar os fenômenos ambientais. Trata sobre a arquitetura do processo descritivo e sua aplicação. A parte 4 apresenta como a aplicação da descrição atua no domínio do conhecimento, abrangendo o domínio da aplicação ou ambiente onde são definidos os requisitos, o domínio da máquina onde são definidos os programas e, consequentemente obtidas entre aplicação e máquina. A título de ilustração dos conceitos, utilizam-se os requisitos de um sistema de gerenciamento de segurança e paisagismo da malha rodoviária na área de transportes. A parte 5 apresenta um resumo da contribuição à área de pesquisa, na organização das idéias e na tecnologia da descrição. A parte 6 apresenta um resumo dos resultados esperados com a aplicação dos fundamentos conceituais da descrição na Engenharia de Requisitos. A parte 7 apresenta um resumo do estado da arte da tecnologia de Engenharia de Requisitos, relacionando algumas bibliografias e um resumo. A parte 8 apresenta idéia de ligação para trabalho futuro, sob o enfoque de inclusão de métricas para avaliação de qualidade dos requisitos desde o início do processo de extração. A parte 9 finaliza um resumo de considerações finais sobre a arte e a técnica da descrição.

2. Contextualização do Problema

Um problema inerente ao processo descritivo é a ambigüidade existente nas declarações de demanda de necessidades e de desejos dos clientes. Quanto maior o número de requisitantes, tanto mais complexo encontrar um termo comum para representar os diferentes pontos de vista e as intenções, observadas as variações dos cenários onde ocorrem os fenômenos do mundo real. Para reduzir o índice de ambigüidade é essencial a resposta a perguntas do tipo o que é, por que e para qual finalidade, a fim de que esteja bem

caracterizado um produto e a certeza do que se trata.

A orientação de esforços na área de Engenharia de Requisitos é pela adoção de uma linguagem que documente precisamente os variados aspectos da extração de requisitos pelo desenvolvedor de software.

O ambiente onde os fenômenos do mundo real ocorrem é de caráter informal e aplicar um formalismo que possibilite a avaliação dos requisitos, sobre consistência de conteúdo e de forma, torna a tarefa muito complexa. Há necessidade de se estabelecer critérios para a descrição de requisitos do cliente quanto à natureza dos requisitos, como parte essencial do estudo do problema. O uso das técnicas deve ter como enfoque básico a avaliação dos requisitos, via análise e revisão de parâmetros e métricas aplicáveis a todo o processo de extração.

Do ponto de vista do desenvolvedor de software, a premissa básica da extração requisitos fundamenta-se em:

- necessidade de adequação de método, na representação do conhecimento;

- os requisitos devem refletir as reais necessidades e desejos do cliente;

- a atividade de captação de requisitos é dependente da fonte de informação, sob os mais variados pontos de vista e de cenários e intenções do cliente;

- os requisitos devem ser consistentes, completos e precisos;

- o domínio do conhecimento é prerrogativa do cliente;

- a atividade de análise do problema para definição de alternativas de solução é sobrecarregada pela deficiência de informação na fase de extração de requisitos;

- do cliente é mais fácil obter informação de como faz e como quer, ao invés de: o que faz, porque o faz e para qual finalidade;

- capitalizar o esforço despendido na extração dos requisitos para posterior aplicação na solução, obtendo a descrição da forma mais completa possível.

Do ponto de vista do cliente, as variáveis condicionantes do processo de descoberta de requisitos que conseqüentemente se tornam impeditivas,

caraterizam-se por:

- o tempo disponível para discussão do problema e da proposta de solução é caro e raro;

- a expectativa de solução do problema é sempre a curto prazo;

- a expectativa com o resultado é sempre acompanhar as novidades tecnológicas;

- a solução do problema com tecnologia da informação implica também a solução do negócio.

O sucesso de implementação do sistema de software e a satisfação dos requisitos do cliente é diretamente proporcional à qualidade do trabalho de extração dos requisitos. A percepção dos fatos e a precisão na análise da natureza dos relacionamentos entre os requisitos, é fundamental em qualquer processo de solução de problemas.

O questionamento dos fatos e a análise de veracidade dos mesmos deve ser uma condição básica do processo de extração de requisitos, comportando um ciclo completo de discussão, com profundidade e exaustão do assunto, preferencialmente, sem pendências para decisão a posteriori. Fundamento este, aplicável à fronteira conhecida da solução do problema, previamente determinada pelo foco e abrangência do domínio da aplicação.

3. Descrição: arte e técnica no descobrimento e comunicação de requisitos

"*Descrever é arte, é saber e ter perícia ao utilizar idéias e executá-las na prática. Descrever é técnica, é ter domínio do processo para melhor executar a arte.*"

A todo momento, o indivíduo defronta-se com a tarefa de representar, na forma falada ou escrita o que quer comunicar, no atendimento às necessidades físicas, emocionais, psíquicas, sociais, econômicas, enfim, existenciais no mundo em que vive. Este comunicar é o fato de tornar comum o entendimento recíproco da idéia entre emissor e receptor. O sucesso nem sempre é obtido nessa comunicação, podendo em sua maioria, a causa ser o não entendimento da mensagem pelo receptor ou, ocorrer equívoco de destinatário.

Na Engenharia de Software, com o processo de representação do conhecimento não é diferente. As diferentes formas de linguagens de comunicação, quer sejam escritas ou gráficas, dependem fundamentalmente do domínio do conhecimento

e de uma série de regras comuns, para se produzir mensagens comunicativas. Representar o mundo real, o ambiente para o qual queira se construir uma solução, requer o domínio de técnicas de descrição por parte do desenvolvedor.

É fato incontestável e admitido por uma grande maioria de desenvolvedores de software, a dificuldade de tratar o detalhamento da representação do conhecimento. Isto é justificado pela falta de habilidade na formulação da escrita dos fenômenos que ocorrem no ambiente em estudo. É mais fácil e rápido partir para uma fase de criar modelos, projetar protótipos de software e tentar a comunicação do resultado com o cliente, do que aprofundar o conteúdo com o conhecimento do domínio do problema ou da demanda requerida. Isto custa tempo e recurso e implica em negociação complexa com o cliente, porque também depende da disponibilidade e do comprometimento do mesmo. A não ocorrência desta fase implica na queima de etapas, o que não permite a maturidade para a ação de proposta de alternativas de solução.

O ambiente é a porção relevante do mundo real para o projeto de desenvolvimento de software. O sistema de software é uma parte da solução baseada em computador que será construída e conectada ao ambiente, como resultado do projeto de desenvolvimento de software[2].

Daí, o repensar na aplicação de maior esforço de detalhamento da demanda requerida pelo cliente, no entendimento e na representação do conhecimento, orientado para o foco e abrangência do problema em estudo.

Em engenharia de requisitos, descrever é arte e implica no domínio de técnicas aplicáveis no processo de descoberta de conhecimento e a conseqüente transformação de idéias.

3.1. Características da Descrição

A descrição é a atividade central do desenvolvimento de software e o meio externamente visível de expressar o pensamento[2]. Apresenta-se na forma escrita ou gráfica para expor, narrar, contar fatos e fenômenos.

A razão fundamental é a representação de idéias e de ações e, não é exagero dizer que o negócio inteiro de desenvolvimento de software é fazer descrições[2].

A descrição fundamenta-se em dois aspectos essenciais: o escopo ou alvo e a abrangência de tempo e de espaço do objeto descrito.

O alvo da descrição é sobre "o que" e "qual" fenômeno do domínio está sendo descrito[2]. A abrangência da descrição é a parte do mundo onde ela está

contida, acerca do "por que" e "para que". Quando se fixa um alvo de descrição está se colocando um prisma através do qual pode-se ver certos conjuntos de fenômenos do domínio. O foco é sobre o domínio completo, mas a visão é muito seletiva. Quando se fixa a abrangência, usualmente focaliza-se a atenção mais narrativamente[2].

Escolher a abrangência certa para uma descrição é muito importante. Isto feito, o que se deseja dizer adapta-se exatamente dentro daquele espaço e tempo. Uma abrangência pequena pode não permitir dizer tudo o que se deseja e, uma abrangência grande pode forçar a dizer muito também. Dizer-se muito em uma descrição faz com que a mesma seja menos independente do contexto que é necessário. Isto pode acarretar menos reuso e pode também frustar uma separação apropriada de conceitos. A correta abrangência não é sempre óbvia. A escolha depende das regras do negócio[2].

Para uma descrição ser útil, seu alvo deve ser conhecido. Primeiro, deve ser conhecido qual domínio, qual parte do universo de interesse está sendo descrito[2]. Se está se olhando um mapa, necessita-se conhecer a representatividade geográfica deste mapa. E, a seguir, qual fenômeno do domínio está sendo descrito. É um mapa de rodovias ou de produção agrícola? Diz-se qual fenômeno está começando a ser descrito pela associação de um conjunto de designações de fenômenos com a técnica de descrição. Para um mapa de rodovias, pode-se designar um fenômeno semelhante a "volume de tráfego de transporte de safra agrícola no trecho rodoviário da área A é v"; para um mapa de produção agrícola, pode-se designar um fenômeno semelhante a "produção de milho por metro quadrado da área B é p".

Para desenvolver sistemas de alguma significância, deve-se fazer muitas descrições. Cada uma será uma descrição parcial, isto é, representará somente uma parte do conceito total, descreverá uma parte do universo completo de interesse.

Adicionando-se uma descrição parcial, após outra, o mundo é começado a descrever pelo incremento narrativo, a cada descrição sucessiva. Evidentemente, quando todas as descrições parciais tenham sido completadas, o resultado final descreve exatamente o mundo de interesse: nem mais, nem menos. Em resumo, uma separação bem sucedida em descrições parciais deve estar baseada em separação de diferentes partes e aspectos do problema[2].

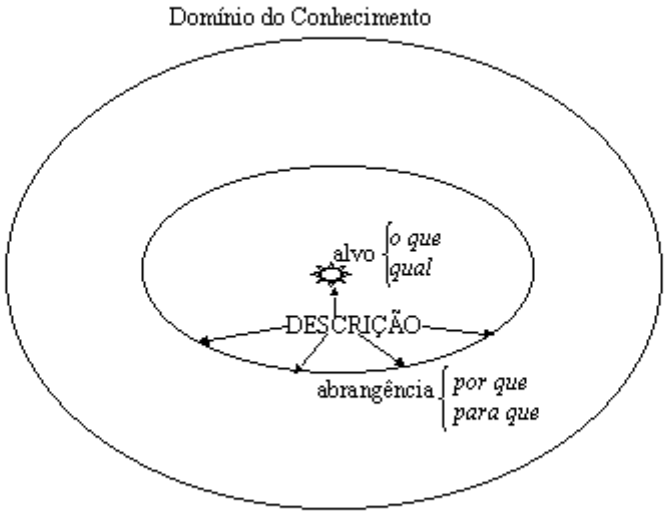


fig.1 - representação do domínio do conhecimento

3.2. Tipos de Descrição

Uma descrição estabelece algum relacionamento existente ou requerido sobre certo fenômeno. Por fenômeno, entende-se tudo o que é percebido pelos sentidos ou pela consciência, considerado dentro de um contexto de domínio do conhecimento. Para entender quais os meios de descrição no domínio, deve-se estar habilitado a identificar o fenômeno.

O primeiro passo no entendimento de descrições é reconhecer que existem quatro importantes tipos de descrições[2]:

- designação;

- definição;

- descrição refutável;

- esboço rudimentar.

3.2.1. Designação

A designação é a atividade de indicar, mostrar, dar a conhecer, simbolizar e classificar fenômenos no domínio do conhecimento. Identificar as características dos fenômenos. É uma descrição informal da reunião de um termo atômico referindo-se ao ambiente[5]. Uma designação simples separa algum conjunto

particular de fenômenos que é de interesse; conta informalmente, em linguagem natural, como reconhecê-lo e dá um nome pelo qual será reconhecido. A

dade precedente à

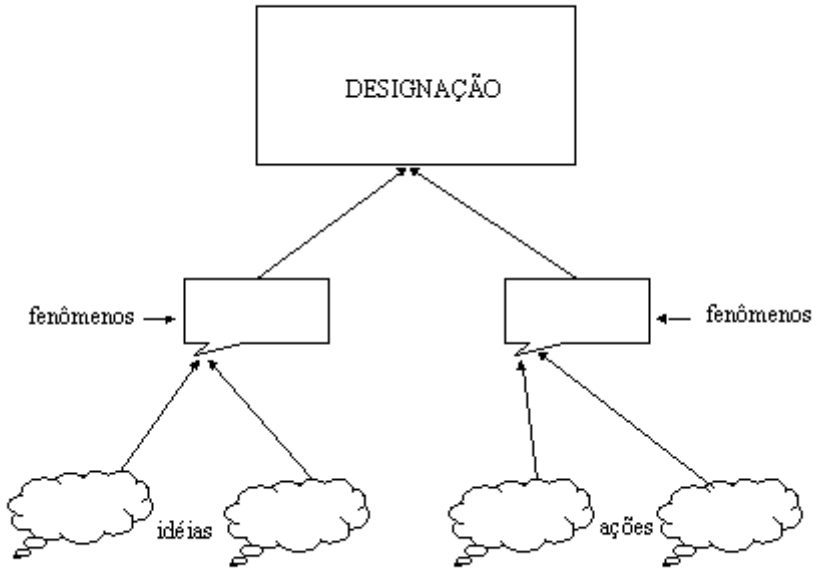


fig.2 - ilustração da designação de fenômenos

A essência de uma boa designação são os fenômenos que são reconhecidos claramente e não ambiguamente no domínio e, a aplicação das regras de reconhecimento que contam o bastante como reconhecê-los. Uma designação é sempre informal, porque o mundo onde os fenômenos são reconhecidos é informal. E, para tal não demandam precisão perfeita em regras de reconhecimento.

Um conjunto de designação não é comum como um dicionário de dados, uma enciclopédia de dados ou repositório. Apresenta três características importantes[2]:

- primeiro, refere-se aos fenômenos designados reconhecíveis no domínio;

- segundo, o objetivo é construir uma ponte entre o domínio do conhecimento e suas descrições, escrevendo designações para os fenômenos essenciais;

- terceiro, um conjunto de designação é local, no sentido que está

associado com algum domínio particular, mas não é amarrado a descrições particulares. Diferentes descrições podem ser aplicadas com o mesmo conjunto de designação para descrever diferentes relacionamentos acerca do mesmo fenômeno.

Como John Von Neumann observou em A Teoria dos Jogos, ‘não existe razão no uso de métodos exatos, onde inexiste claridade nos conceitos e assuntos’. Ou melhor, ‘Não existe sentido em se começar algo, precisamente, quando não se conhece acerca do que se está tratando’[2].

3.2.2. Definição

A definição é a atividade de enunciar os atributos essenciais e específicos de um objeto ou fenômeno. É uma descrição formal de um termo atômico, usando outros termos definidos ou designados[5].

Em desenvolvimento de software é necessário suprir um conjunto de definições, uma para cada palavra difícil, quando se está descrevendo um domínio.

Uma definição dá uma qualificação formal de um termo que pode ser usado por outra descrição. É expressada sobre termos definidos previamente. Uma definição não pode ser verdadeira ou falsa, ela pode ser somente bem ou mal formulada e ser útil ou não útil[2].

Cada termo usado na descrição deve estar definido em algum lugar. Esclarece, se os leitores não conhecem acerca do que se está falando e garante ao redator, a certeza do que está expressando.

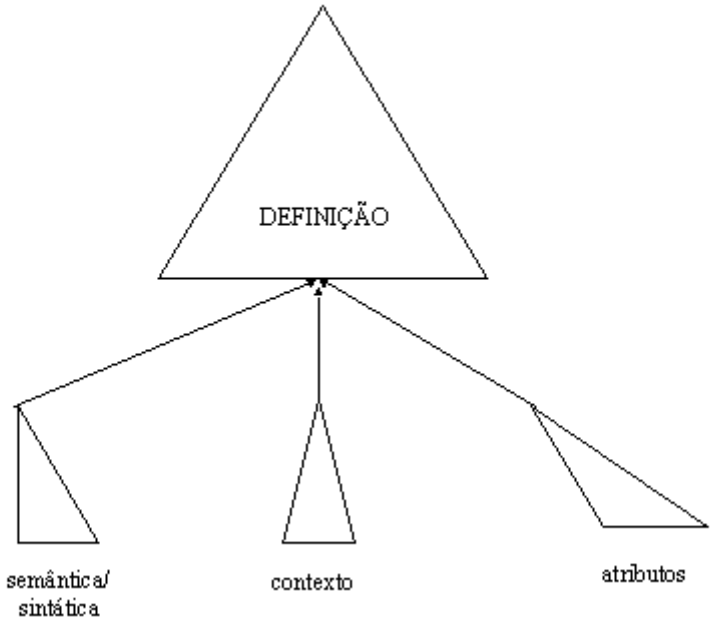


fig.3 - ilustração da definição dos fenômenos

3.2.3. Descrição Refutável

Uma descrição refutável descreve sobre domínio, dizendo algo acerca do que poderia, em princípio, ser refutado ou desaprovado. Refutabilidade depende do fato de a descrição apresentada referir-se a fenômeno do domínio descrito.

Refutar uma afirmação é demonstrar que ela está errada. Não é justo afirmar que está errado, só para acrescentar uma diferença e afirmação de competição, mas para demonstrar claramente que ela não se faz adequada ao fato[2].

Respeitáveis teorias científicas são refutáveis. Um cientista que disponibiliza antecipadamente uma teoria, corre o risco que outro cientista pontue algum conhecimento do fato ou execute um experimento, demonstrando que a teoria está errada. Esta exposição ao risco de refutação é o grande forte das ciências naturais estabelecidas. Respeitáveis desenvol-vedores de software dependem de refutabilidade de forma similar:

- uma descrição de domínio do ambiente do sistema ou de domínio da aplicação afirma como são os pensamentos;
- uma descrição de requisito afirma descrever como os pensamentos devem ser para quando o sistema seja instalado e operado.

Ambos devem ser refutáveis. A descrição de domínio deve processar o risco que alguns poderão dizer: 'Aquilo não é verdade - existe um contra-exemplo'. A

descrição de requisito deve processar dois riscos, em que o cliente dirá: 'Não, este não é o efeito requerido', ou então: 'Sim, isto é o efeito requerido, mas o sistema não está realizando algo, olhando para o que ele descreve aqui'[2].

Para atingir refutabilidade é necessário explicitar designações. Escreve-se uma designação a cada conjunto básico de fenômeno, descrevendo como reconhecê-lo no domínio e dá-se o nome que será usado na descrição. A escolha de fenômeno e designações é severamente restritiva: o fenômeno deve ser com

:e porque possibilita

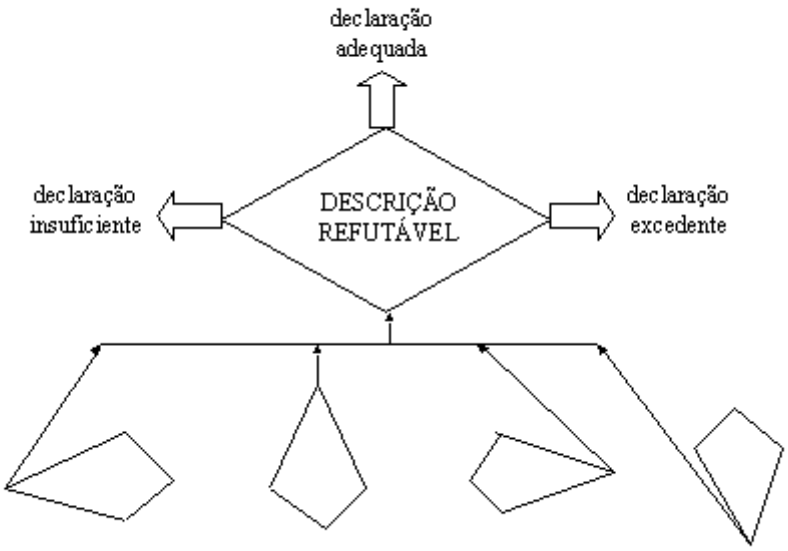


fig.4 - ilustração da descrição refutável

3.2.4. Esboço Rudimentar

Um esboço rudimentar é uma tentativa de descrição de algo que está em processo de exploração ou de invenção. A definição característica de um esboço rudimentar é sua imprecisão. Esta imprecisão é inevitável quando se desenvolve o foco inicial.

Usa termos indefinidos para registrar idéias vagas e mal formuladas. Isto registra imprecisão, idéias meio formadas, para um estágio quando se deseja dizer: ‘bem, algo rudimentarmente igual a isto pode estar certo’, sem devotar muito tempo ou esforço para dizer exatamente o que se tem em mente. Pode-se ter uma idéia rudimentar e esboçar rapidamente antes que se esqueça. Muita precisão poderá ser prejudicial, não útil no momento da captação e poderá inibir

o fluxo de idéias[2].

O esboço rudimentar tem uma parte apropriada em desenvolvimento de software, mas é limitada. Ele pertence somente à ligação preliminar de cada estágio do desenvolvimento. Podem ser esboços de requisitos, de arquitetura de sistema, de ambiente de sistemas e de programas. Em cada caso devem progredir para a descrição precisa. Deve-se conservar o esboço somente para a consulta sobre o caminho do desenvolvimento ou motivação ou, para retrabalhá-lo em descrição precisa ou, descartá-lo como andaime de construção após mentação precisa.

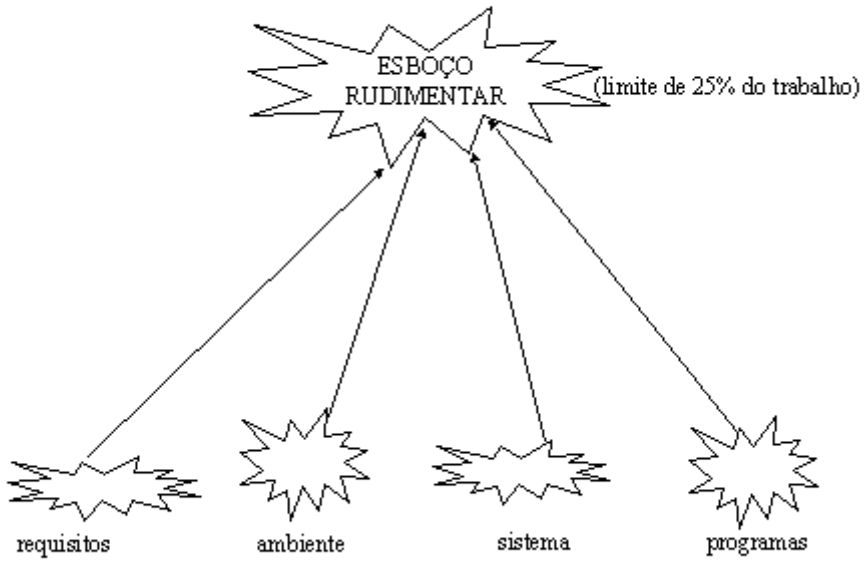


fig.5 - ilustração do esboço rudimentar

Se o esboço usurpar o papel da descrição precisa, está se fazendo um grande desserviço. Deve-se sempre estar em guarda contra o perigo de se produzir um esboço quando algo diferente é necessário. Se mais que 25% da descrição são esboços, então o projeto é trivial e não importante, não se está fazendo a coisa correta[2].

3.3. Tecnologia e Arquitetura da Descrição

A tecnologia da descrição requer uma abordagem específica para cada caso em estudo. Inicialmente, utiliza-se da designação para dar a conhecer o fenômeno e simbolizá-lo, para numa fase seguinte, classificá-lo na categoria dos fenômenos descritos.

Na seqüência, a definição conceitual formará uma base sólida de referência aos atributos dos fenômenos. A etapa seguinte deverá observar o alvo da descrição, o que norteará o foco de estudo e, em paralelo é necessário identificar a abrangência da descrição.

Para se chegar ao nível de qualidade exigido de uma descrição precisa, que é o objetivo final, o uso de tipos intermediários de descrição irá auxiliar no processo de desenvolvimento. Descrições parciais e esbocos fazem parte de uma

avés de refinamentos

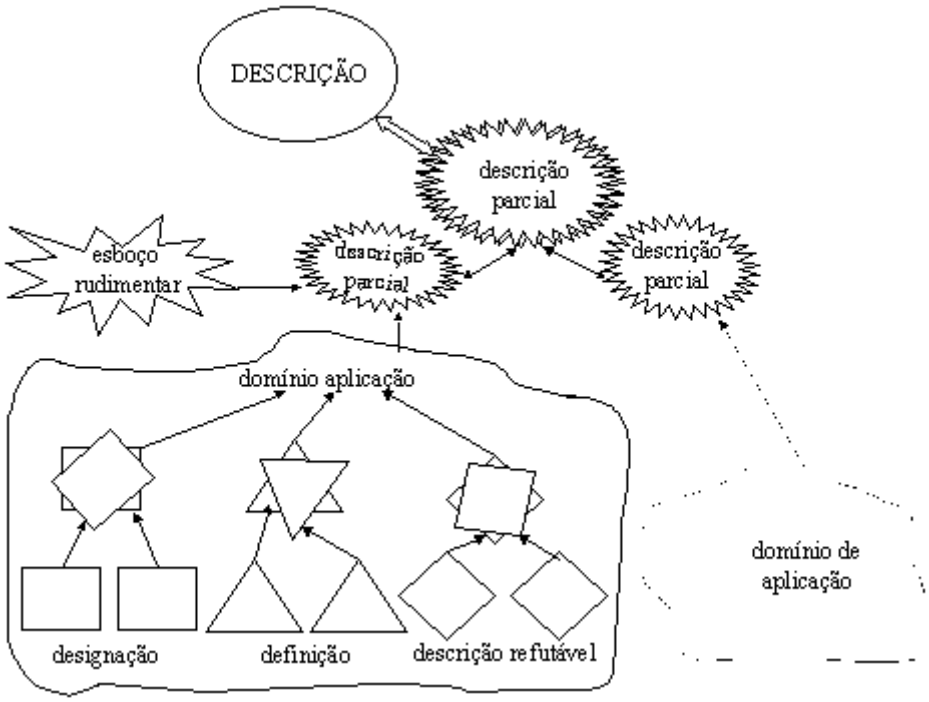


fig.6 - ilustração da arquitetura da descrição

4. Aplicação da Descrição na Representação do Domínio do Conhecimento

Em cada atividade de desenvolvimento de software, o contexto do problema tem, ao menos, dois domínios distintos: o domínio da aplicação (o ambiente ou o mundo real ou matéria julgada) e o software. O domínio da aplicação é onde os requisitos do cliente existem; o software sustenta uma solução para o problema

por interação em alguns caminhos com o domínio da aplicação. Pode-se pensar do domínio da aplicação como o que é dado e o domínio do software como o que será construído[2].

A aplicação das técnicas de descrição são evidenciadas no detalhamento do ecificações de .

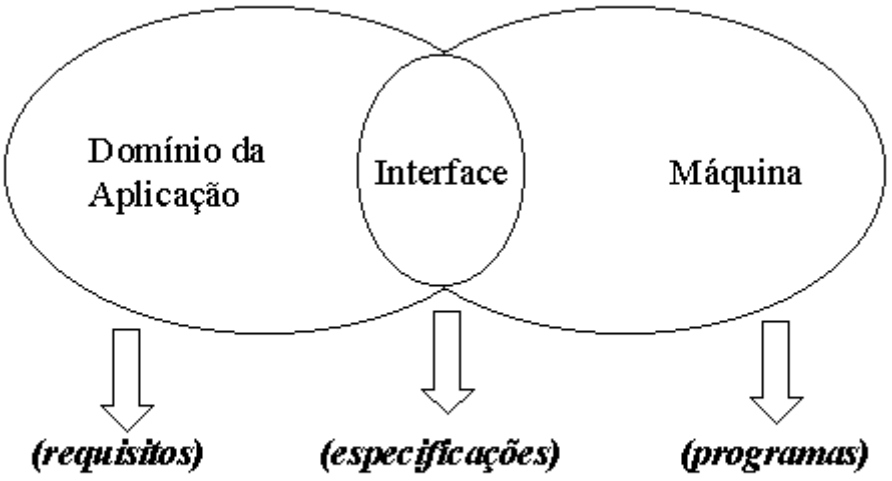


fig.7 - representação do domínio do conhecimento

Exemplo:

A proposta de aplicação de exemplo ilustrativo, apresenta um estudo feito dentro do sistema de administração de transportes do governo do Estado do Paraná, mais especificamente o rodoviário, do qual o Departamento de Estradas de Rodagem é o órgão responsável. Dentre os diversos domínios de aplicação, foi escolhido para ilustração do estudo, o sistema de gestão de segurança e paisagismo da malha rodoviária, a motivação pela escolha é justificada pelo caráter de impacto social.

4.1. Domínio da Aplicação ou Ambiente

A parte do mundo em que os efeitos da máquina são sentidos, avaliados e, se são sucesso, aprovados pelo cliente, é o domínio da aplicação. O uso do termo domínio da aplicação é aplicado pelo autor[2] para especificar um problema específico. Não um domínio genérico denotando classes de aplicações, tais como

domínio bancário, domínio de chaveamento de telefone, etc..., mas um domínio particular de âmbito menor, por exemplo: de empregados de uma companhia, a superfície de controle particular de aviões, seus suprimentos, produtos e encomendas.

O primeiro passo em análise de problemas é estruturar e analisar o domínio da aplicação. O princípio de relevância do domínio, declara que: "Tudo o que é relevante para os requisitos, deve aparecer em alguma parte do domínio da aplicação". Existe uma importante conseqüência deste princípio. O domínio da aplicação não é limitado a partes do mundo que estão diretamente ligados à máquina. É mais abrangente, refere-se ao negócio[2].

O domínio da aplicação é onde os requisitos particulares dos clientes são encontrados. Se o domínio da aplicação não for identificado corretamente não se está apto para focalizar os requisitos dos clientes.

Exemplo: malha rodoviária

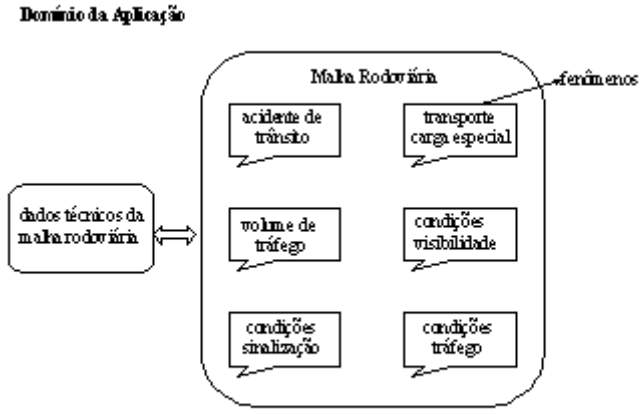


fig.8 - exemplo representação do domínio da aplicação

O domínio da aplicação do sistema de gestão de segurança da malha rodoviária tem como alvo gerenciar os fenômenos que são relevantes e que constituem preocupação do corpo gerencial técnico de segurança rodoviária e ocorrem no contexto da malha rodoviária, nos limites do estado.

4.2. Requisitos no Domínio da Aplicação

Requisitos são fenômenos acerca do domínio da aplicação. Para descrevê-los exatamente, descrevem-se os relacionamentos requeridos entre os fenômenos

do contexto do problema[2].

Muitos projetos falham porque seus requisitos estão inadequadamente explorados e descritos. Os requisitos estão localizados no domínio da aplicação, onde está o problema, mas em muitos desenvolvimentos, toda documentação e discussão sérias focam sobre a máquina que é oferecida como uma solução para o problema[2].

O requisito é uma propriedade do domínio da aplicação que o sistema de software deve executar. A prática tradicional em desenvolvimento de software tem sido ignorar o domínio da aplicação e focalizar atenção sobre a máquina. Isto é porque alguns escritores têm tentado explicar a distinção entre o que e como, tal como uma distinção entre fora e dentro da máquina ou entre uma descrição lógica e uma descrição física da máquina ou, entre um sistema essencial e um sistema implementável, isto é, entre a máquina como poderá ser com tecnologia perfeita e a máquina tal como será na implementação real. [2]. Não surpreendentemente quando se focaliza a atenção sobre a máquina, desta forma constata-se que os clientes não são muito bons para decidir sobre seus requisitos. Eles não conhecem acerca de computadores e, portanto, não podem escolher sobre a máquina que necessitam. Tradicionalmente, os desenvolvedores de software têm pensado de seus clientes como ursos de cérebro muito pequeno. "Eles não sabem o que querem", é o que se ouve. Mas, na verdade, esta postura de perspectiva superior é reflexo das próprias limitações e falhas dos desenvolvedores[2].

Exemplo: malha rodoviária

Os requisitos do domínio da aplicação do sistema de gestão de segurança da malha rodoviária são decorrentes dos fenômenos que são relevantes na vocação do órgão relativo à missão técnica de garantia de segurança de trânsito na malha rodoviária do estado. Neste particular estão delineadas as declarações de necessidades sob o ponto de vista de cada informante e/ou agente do processo produtivo ou, mesmo, usuário da informação.

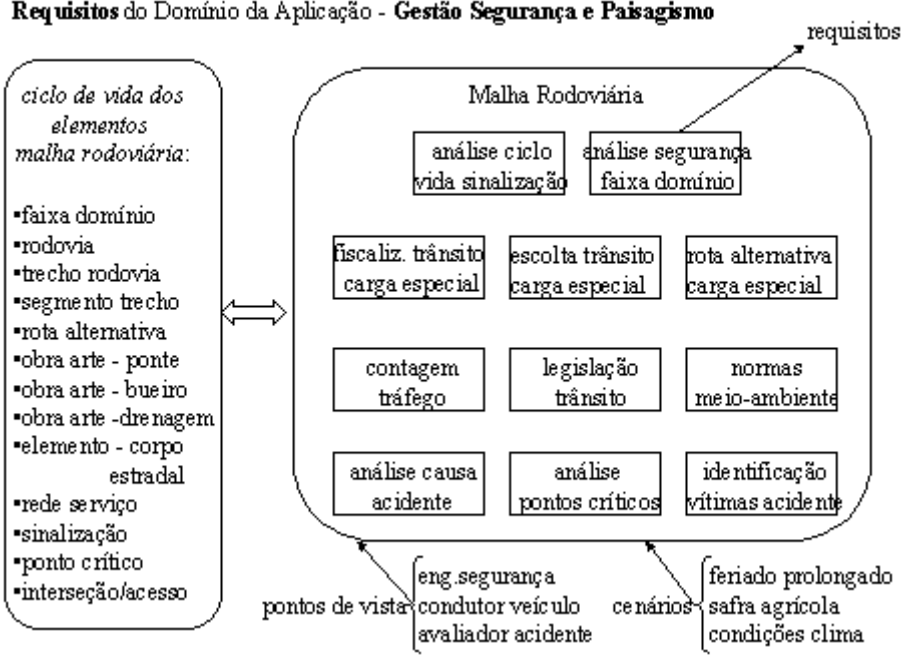


fig.9 - exemplo representação dos requisitos

4.3. Especificações

Especificação é uma propriedade, dirigida a ser implementável para suportar a satisfação do requisito.

A terminologia de desenvolvimento de software não é padronizada e o conceito de especificação é um dos elementos que fazem parte do universo do contexto da engenharia de software. Dentre eles o autor[2] selecionou a que mais se aproxima e restringe a forma de pensar, referenciando Carlo Ghezzi, em Fundamentos de Engenharia de Software:

"*Em geral, pode-se ver uma especificação como uma declaração de acordo entre um produtor e um consumidor de serviço ou um implantador e um usuário*".

Especificações são coisas difíceis. Embora sejam requisitos de um tipo, podem não serem vistas para fazer sentido óbvio no ambiente. Isto porque são derivados dos requisitos do cliente por um número de passos de raciocínio. Especificações representam a resposta das questões: Qual comportamento que a especificação de interface poder produzir os efeitos que são requeridos no ambiente? O que poderá a máquina requerer para produzir uma saída particular em resposta a determinada entrada? Qual comportamento para a especificação de interface deve o programa produzir?...

Exemplo: malha rodoviária

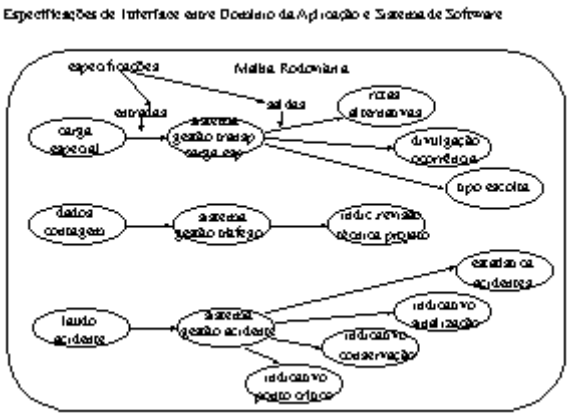


fig.10 - exemplo representação especificações

As especificações dos fenômenos de interface entre o domínio da aplicação e o sistema de software referem-se aos procedimentos de entrada de informações para o sistema e aos resultados processados pelo sistema de software.

5. Contribuição na Área de Engenharia de Software

A contribuição fundamental para a pesquisa na área de engenharia de requisitos é a visão conceitual da descrição como arte e técnica no descobrimento e na documentação do conhecimento e representação de requisitos. Deve ser utilizada extensiva e intensivamente até exaurir possibilidades de dúvida e discordâncias e, especialmente, ambigüidades. A descrição é uma tarefa árdua, trabalhosa e complexa, muitas vezes incompreendida por pessoas alheias ao processo de engenharia de software, especificamente o cliente contratante da solução do problema, cuja expectativa de resultado é imediatista.

O processo de análise dos requisitos não deve ser uma fato posterior à extração

de requisitos e sim concomitante. A ordem não é representar, modelar, para depois ver para que serve e sim, descrever o mais detalhadamente possível os fenômenos do ambiente que caracterizam o problema e, moldar uma solução de sistema de software com critérios de viabilidade de implementação, antes do início do desenvolvimento do software.

6. Resultados Esperados

A expectativa com o resultado deste trabalho é ordenar conceitualmente a atividade descritiva do processo de descoberta de conhecimento, em etapas sucessivas e complementares, alimentadas de forma recursiva. Especialmente, em promover o espírito crítico e a real necessidade de se reforçar o domínio do conhecimento do ambiente em estudo e os requisitos. Com isto, além de documentar o processo descritivo com o foco no problema, propiciar o preenchimento de uma lacuna na fase inicial de extração de requisitos, especificações e o domínio do conhecimento que não é coberta em sua totalidade pelas metodologias e linguagens de modelagem.

Muitas técnicas auxiliam a documentação, entretanto, o uso adequado de uma ferramenta, não a qualifica como infalível nos critérios de validação de requisitos de engenharia de software.

7. O Estado da Arte

Quanto ao foco do tema do trabalho, a fonte de pesquisa ateve-se à base de informações [2] e [5]. Em paralelo foram realizadas pesquisas sobre a aplicabilidade de validação do conteúdo dos requisitos extraídos e tem-se a relatar o seguinte:

Em [1], o artigo descreve uma técnica de análise formal chamada checagem de consistência para detecção automática de erros. A técnica é projetada para analisar especificações de requisitos, expressadas na notação tabular SCR (Software Cost Redution), onde é discutido o papel da checagem de consistência, durante a fase de requisitos do desenvolvimento do software.

Em [3], o autor relata que ferramentas CASE OO podem ajudar no processo de extração de requisitos, mas elas não garantem que os requisitos sejam extraídos corretamente. A necessidade é de uma técnica de engenharia de requisitos que habilite a efetiva descoberta de requisitos. Concluindo, registra que o desenvolvimento de um sistema de qualquer tamanho deve satisfazer a um complexo e variado conjunto de conflito de interesses, emanado de várias e diferentes perspectivas.

Em [4], os autores propõem uma forma para detecção de falhas em documento de requisitos do usuário URD (User Requirements Document), uma descrição do usuário da funcionalidade e performance de um produto de software por sustentação de N inspeções formais do documento executadas em paralelo. Os autores partem da hipótese de que a utilização de N equipes de inspeções separadas pode significar multiplicidade de esforço, no entanto, o total de número de falhas detectáveis é bem maior que em uma simples inspeção.

Em [6], a conferência internacional de engenharia de software, reflete o conceito crescente de aliar a pesquisa à prática na engenharia de requisitos. Neste compêndio, estão editados os trabalhos selecionados para a apresentação, dentre os quais dois foram editados concomitantemente na revista IEEE software março/abril 1998. A estrutura dos assuntos em seções permite a pesquisa dos mais variados temas, dentre eles: extração de requisitos, modelagem e requisitos, cenários, intenções, rastreamento e gerenciamento de mudança de requisitos, segurança, sobrevivência e tolerância a falha.

8. Trabalhos Futuros

Criar mecanismo de validação de requisito descrito, a partir da informação do cliente, ajustável aos diversos níveis do detalhamento descritivo. Aplicar um modelo de atribuição de pesos em função da qualificação da influência e do poder de decisão da fonte de informação para eliminar os pontos extremos de ambigüidade. Outro aspecto a considerar é o enfoque da abrangência do uso da informação pelo cliente e para qual finalidade.

9. Conclusão

Na Engenharia de Requisitos, os estudos dos diversos tipos de descrição e suas formas de estruturação, compõem a tecnologia de descrição e a arquitetura utilizada para extração de requisitos e a especificação. O objeto de estudo foi centrado no fundamento conceitual da descrição como atividade central no desenvolvimento de software. A terminologia e sua aplicabilidade compõem uma

primeira fase de estudo sobre o assunto, a representação informal e descritiva do conhecimento aplicável à fase inicial de estudo do problema.

A tecnologia de descrição constitui um ferramental poderoso na representação do conhecimento, desde que observadas regras básicas na estrutura da descrição e a escolha adequada para cada caso.

Referências Bibliográficas

[1] HEITMEYER, Constance L.; JEFFORDS, Ralph D.; LABAW, Bruce G. Automated consistency checking of requirements specifications. ACM, USA, v. 5, n.3, p. 231-261, jul. 1996.

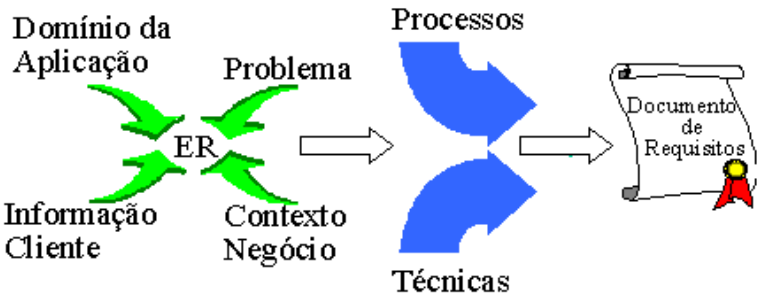
[2] JACKSON, Michael. Software requirements and specifications: a lexicon of practice, principles and prejudices. Massachusetts : Addison-Wesley, 1995. p. 228.

[3] LEWIS, Mark. Getting the requirements. Object Magazine, New York, v. 6, n. 9, p. 42-47, nov. 1996.

[4] SCHNEIDER, G. Michael; MARTIN, Johnny; TSAI, W.T. An experimental study of fault detection in user requirements documents. ACM, USA, v. 1, n.2, p. 188-204, abr. 1992.

[5] ZAVE, Pamela; JACKSON, Michael. Four dark corners of requirements engineering. ACM, USA, v. 6, n.1, p. 1-30, abr. 1997.

[6] ICRE Third International Conference on Requirements Engineering. Proceedings... USA: IEEE Computer Society, 1998, v. 1, p. 250.


Domínio da
Aplicação
Problema
ER
Informação
Cliente
Contexto
Negócio
Processos
Técnicas
Documento
de
Requisitos