

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/229042029>

# Uma Ontologia para o Domínio de Qualidade de Software com Foco em Produtos e Processos de Software

**Article** · January 2008

CITATIONS

2

READS

99

**3 authors**, including:



**Rodrigo Dal Moro**

Universidade Federal do Espírito Santo

**7** PUBLICATIONS **68** CITATIONS

[SEE PROFILE](#)



**Ricardo de Almeida Falbo**

Universidade Federal do Espírito Santo

**172** PUBLICATIONS **1,661** CITATIONS

[SEE PROFILE](#)

**Some of the authors of this publication are also working on these related projects:**



INTEROPERABILIDADE SEMÂNTICA DE INFORMAÇÕES EM SEGURANÇA PÚBLICA [View project](#)



Knowledge Management in Software Testing [View project](#)

## Uma Ontologia para o Domínio de Qualidade de Software com Foco em Produtos e Processos de Software

Rodrigo Dal Moro, Ricardo de Almeida Falbo

Mestrado em Informática – Universidade Federal do Espírito Santo (UFES)  
Av. Fernando Ferrari s/n, Campus de Goiabeiras – 29.060-900 – Vitória – ES – Brasil

rdalmoro@yahoo.com.br, falbo@inf.ufes.br

***Abstract.** Software quality involves several perspectives, being the main two software process and product quality. Although there are very different aspects in each perspective, it is also possible to identify commonalities. In fact, in both cases, we are talking about entities that must be measured to have their quality evaluated. This paper presents an ontology about the Software Quality domain that was developed aiming at partially formalizing the knowledge involved in it, with focus on common aspects of process and product quality.*

***Resumo.** Qualidade de software envolve diversas perspectivas, sendo as duas principais a qualidade de processo e a qualidade de produtos de software. Ainda que realmente haja aspectos bastante diferentes em cada perspectiva, é possível identificar, também, aspectos comuns. De fato, em ambos os casos, trata-se de entidades que precisam ser mensuradas para ter sua qualidade avaliada. Este artigo apresenta uma ontologia do domínio de Qualidade de Software desenvolvida visando formalizar parcialmente o conhecimento envolvido nesse domínio, focando na captura dos aspectos comuns às perspectivas de qualidade de processos e de produtos de software.*

### 1. Introdução

É fato que a qualidade dos produtos de software depende fortemente da qualidade dos processos utilizados em seu desenvolvimento e manutenção. Assim, a busca pela qualidade em software passa tanto pela qualidade dos processos quanto pela de produtos de software. No caso de produtos, é necessário avaliar, dentre outros, conformidade a padrões, atendimento a requisitos e consistência com outros artefatos produzidos no processo de software. Em relação a processos de software, o foco deve ser a melhoria contínua.

Uma parte importante da garantia da qualidade é a avaliação. A avaliação sistemática da qualidade de processos, seus ativos (atividades, ferramentas, procedimentos e recursos) e dos artefatos produzidos é essencial para o bom andamento de um projeto e para apoiar a implementação de estratégias de melhoria [Fuggetta 2000]. Contudo, uma avaliação sistemática não pode ser conduzida em bases meramente subjetivas. É necessário identificar características que sejam capazes de indicar a qualidade de uma entidade, medir essas características, analisar os resultados das medições e concluir sobre a necessidade de ajustes. Pode-se notar, portanto, que a

medição e a avaliação têm um papel fundamental para se conhecer a qualidade dos processos e produtos de software de uma organização.

Entretanto, o estudo acerca da qualidade de software tem sido sempre bastante segmentado de acordo com essas duas perspectivas: qualidade de processos e qualidade de produtos. Por exemplo, há diversas normas e modelos de qualidade, tais como CMMI-SW [SEI 2006], ISO/IEC 12207 [ISO/IEC 2008], ISO/IEC 15504 [ISO/IEC 2003] e MPS.BR [Softex, 2007], focando aspectos relacionados à qualidade de processos de software, enquanto outros focam a qualidade de produtos de software, tais como as normas ISO/IEC 9126 [ISO/IEC 2001] e a ISO/IEC 14598 [ISO/IEC 1999].

Ainda que realmente haja aspectos bastante diferentes em cada perspectiva, é possível identificar, também, aspectos comuns. De fato, em ambos os casos, trata-se de entidades que precisam ser mensuradas para ter sua qualidade avaliada. Assim, é importante tentar prover uma conceituação básica acerca de qualidade de software que se aplique tanto a produtos quanto a processos de software. Isso pode ser útil, por exemplo, para o desenvolvimento e integração de ferramentas de apoio à garantia da qualidade, avaliação e melhoria de processos e produtos de software.

Este artigo apresenta uma ontologia do domínio de Qualidade de Software, visando formalizar parcialmente o conhecimento envolvido nesse domínio, tendo como foco principal a captura dos aspectos comuns às perspectivas de produto e processo. Dada a complexidade do domínio, a ontologia foi desenvolvida em três fragmentos: estrutura de modelos de qualidade, medição e avaliação. Devido às limitações de espaço, neste artigo apenas os dois primeiros fragmentos, as sub-ontologias de estrutura de modelos de qualidade e medição, são apresentados.

O artigo está estruturado como se segue. A Seção 2 trata brevemente do universo de discurso da ontologia proposta, qualidade de software, e discute algumas ontologias existentes para esse domínio. A Seção 3 apresenta as sub-ontologias de estrutura de modelos de qualidade e medição desenvolvidas, bem como compara a ontologia aqui proposta com a descrita em [Bertoa et al. 2006]. Finalmente, a Seção 4 apresenta as considerações finais do trabalho.

## **2. Qualidade de Software e Ontologias**

Qualidade do produto de software pode ser vista como um conjunto de características a serem satisfeitas em um determinado grau, de modo que o produto atenda às necessidades explícitas e implícitas de seus usuários [Rocha et al. 2001]. Entretanto, não se obtém qualidade do produto de forma espontânea. Ela tem de ser construída. Assim, a qualidade do produto depende fortemente da qualidade do processo seguido durante o seu desenvolvimento [Fuggetta 2000]. De fato, qualidade de software envolve diversas perspectivas, tais como qualidade de processos, de produtos e de recursos.

A área de qualidade de software tem se desenvolvido bastante pautada na dualidade produto-processo. De maneira geral, normas e modelos de qualidade de processo definem boas práticas que devem estar presentes em processos ou resultados esperados dos mesmos. O modelo CMMI-SW [SEI 2006], por exemplo, define áreas de processo e metas e práticas que devem ser atingidas e realizadas nessas áreas. A norma ISO/IEC 12207 [ISO/IEC 2008], por sua vez, define um conjunto de processos

associados a propósitos e resultados esperados, bem como atividades e tarefas a serem implementadas em cada processo. Há ainda trabalhos relacionados a como avaliar processos, tal como a norma ISO/IEC 15504 [ISO/IEC 2003]. Essa norma trata da avaliação de processos e como utilizar uma avaliação de processo para determinar a capacidade e melhorar o processo avaliado. Segundo ela, como resultado de uma implementação bem sucedida de uma avaliação de processo, esperam-se determinar informações que caracterizem o processo e em que extensão o processo atinge seu propósito, o que envolve um arcabouço de medição de atributos do processo.

Já normas e modelos de qualidade relacionados a produtos de software focam em características de qualidade que um produto deve apresentar e medidas que podem ser utilizadas para se avaliar tais características, tal como a série ISO/IEC 9126 [ISO/IEC 2001]. De maneira análoga aos processos de software, há também trabalhos relacionados a como avaliar em que grau produtos possuem determinadas características, tal como a série de normas ISO/IEC 14598 [ISO/IEC 1999].

Pode-se perceber que há aspectos bastante diferentes em cada perspectiva. As normas e modelos de qualidade de processo estão focados principalmente em processos, propósitos, resultados e atividades. As normas de qualidade de produto, por sua vez, enfocam características dos produtos e como medi-las. Entretanto, há também aspectos comuns. Ambas as perspectivas reconhecem a importância de se avaliar a qualidade das respectivas entidades (processo ou produto). Ou seja, trata-se de entidades que precisam ser mensuradas para ter sua qualidade avaliada.

Analisando-se normas e métodos de avaliação da qualidade de produtos e processos de software, nota-se que a medição é uma parte essencial da avaliação. É necessário, primeiro, definir que entidades (processos, produtos, recursos) serão objetos de medição e quais de suas características serão medidas. Segundo, é preciso definir quais medidas serão usadas para quantificar essas características. Medidas podem ser básicas, i.e. funcionalmente independentes de outras medidas, ou derivadas, definidas como uma função de outras medidas [ISO/IEC 2002]. Uma vez planejada a medição, pode-se passar efetivamente à sua execução, o que envolve a aplicação de procedimentos de medição para se obter um valor para uma medida da entidade em questão. Uma vez realizada a medição, devem-se analisar seus resultados para avaliar a qualidade das entidades medidas.

Assim, é importante tentar prover uma conceituação básica acerca de qualidade de software que se aplique tanto a produtos quanto a processos de software. Uma conceituação corresponde ao conjunto de conceitos usados para interligar abstrações de entidades de um dado domínio. Uma ontologia é uma descrição formal e explícita de uma conceituação [Guizzardi 2005]. Ao explicitar e formalmente definir os conceitos, relações, propriedades e restrições em um domínio particular, tem-se um artefato de engenharia que expressa a conceituação desse domínio, dita uma ontologia de domínio. Uma ontologia de domínio pode ter diversos usos, tais como [Jasper e Uschold 1999]: (i) apoiar a comunicação entre pessoas trabalhando no domínio; (ii) integração de dados, modelos e sistemas desenvolvidos de forma independente para o domínio, (iii) como uma especificação reutilizável para a construção de sistemas no domínio, (iv) como apoio para manutenção de sistemas, servindo de documentação dos sistemas desenvolvidos para o domínio.

Pensando nesses usos, conclui-se que uma ontologia de qualidade de software é de suma importância para a área. Por exemplo, uma ontologia de qualidade que trate uniformemente produtos e processos pode ser utilizada para o desenvolvimento de ferramentas de apoio a um processo de Garantia da Qualidade, cujo propósito é assegurar que tanto produtos quanto processos estão em conformidade com padrões organizacionais [Softex 2007]. Essa mesma conceituação pode ser usada e estendida para o desenvolvimento de ferramentas de apoio a um processo de Avaliação e Melhoria de Processos, que visa determinar o quanto os processos da organização contribuem para alcançar seus objetivos de negócio, além de apoiar a organização a planejar, realizar e implantar melhorias contínuas nos processos com base no entendimento de seus pontos fortes e fracos [Softex 2007]. De fato, esse uso específico motivou-nos a buscar ontologias propostas para esse domínio, tendo se chegado a três trabalhos principais: [Duarte e Falbo 2000], [Martin e Olsina 2003] e [Bertoa et al. 2006].

Duarte e Falbo (2000) desenvolveram uma ontologia de qualidade de software sub-dividida em duas sub-ontologias: *Características de Qualidade – Natureza e Estrutura*, fortemente baseada na ISO 9126 (versão de 1991) e que trata dos tipos de características de qualidade e sua aplicação; e *Qualidade – Como Medir*, que, de modo simplificado, trata da medição de software.

Martin e Olsina (2003) desenvolveram uma ontologia para métricas e indicadores de software para *Web*, com o objetivo formalizar e homogeneizar a conceituação nesse domínio. Para tal, sua ontologia está centrada em termos das normas internacionais ISO/IEC 15939 [ISO/IEC 2002], ISO/IEC 9126 e ISO/IEC 14598.

Bertoa et al. (2006) desenvolveram uma ontologia para o domínio de medição de software, buscando definir uma terminologia coerente, consensual e amplamente aceita, a partir da análise de diversos trabalhos, padrões e normas. Como resultado dessa análise, foram definidas quatro sub-ontologias: *Caracterização e Objetivos da Medição de Software*, que trata do estabelecimento do contexto e das metas da medição; *Medidas de Software*, que busca clarificar a terminologia envolvida nas definições de medidas; *Abordagens de Medição*, que descreve diferentes modos de obter resultados de medição; e *Medição*, que inclui conceitos relacionados à realização do processo de medição.

Analisando-se esses trabalhos, conclui-se que os dois primeiros não atendem o propósito definido para a ontologia pretendida, tendo em vista que sua visão está muito centrada na perspectiva de produtos de software. Já o trabalho de Bertoa et al. (2006) é mais geral. Grande parte de seus conceitos tem origem na norma ISO/IEC 15939, que é de propósito geral, abrangendo, portanto, tanto a medição de processos e quanto de produtos de software. Contudo, há também muitos conceitos oriundos da ISO/IEC 14598, que é específica para produtos.

Como resultado dessa análise, decidiu-se desenvolver uma ontologia de qualidade com foco em produtos e processos de software. Como qualquer artefato de engenharia, uma ontologia deve ser construída seguindo métodos apropriados. Para desenvolver a ontologia de qualidade de software, foi adotado o método SABiO (*Systematic Approach for Building Ontologies*) [Falbo, 2004].

SABiO define um processo para construção de ontologias, cujas principais atividades são: (i) identificação do propósito e especificação de requisitos, que visa

identificar questões que a ontologia deve ser capaz de responder (questões de competência), (ii) captura da ontologia, que tem por objetivo capturar os conceitos, relações, propriedades e restrições relevantes sobre o domínio em questão; e (iii) formalização, que busca escrever os axiomas da ontologia em uma linguagem formal (neste trabalho, optou-se pela lógica de primeira ordem). Paralelamente a essas atividades, ocorrem as atividades de: (iv) integração com ontologias existentes, que visa reutilizar conceituações existentes e integrar a ontologia em desenvolvimento a uma rede de ontologias mais ampla; (v) avaliação da ontologia, que, dentre outros, trata de avaliar se a ontologia é capaz de responder às questões de competência; e (vi) documentação da ontologia, que visa registrar o desenvolvimento da ontologia.

SABiO advoga, ainda, o uso de uma linguagem de modelagem para facilitar a comunicação dos modelos da ontologia, sugerindo um perfil UML como linguagem gráfica para representação de ontologias [Mian e Falbo, 2003], adotado neste trabalho. A Figura 1 mostra as principais notações desse perfil e alguns dos axiomas definidos para as mesmas.

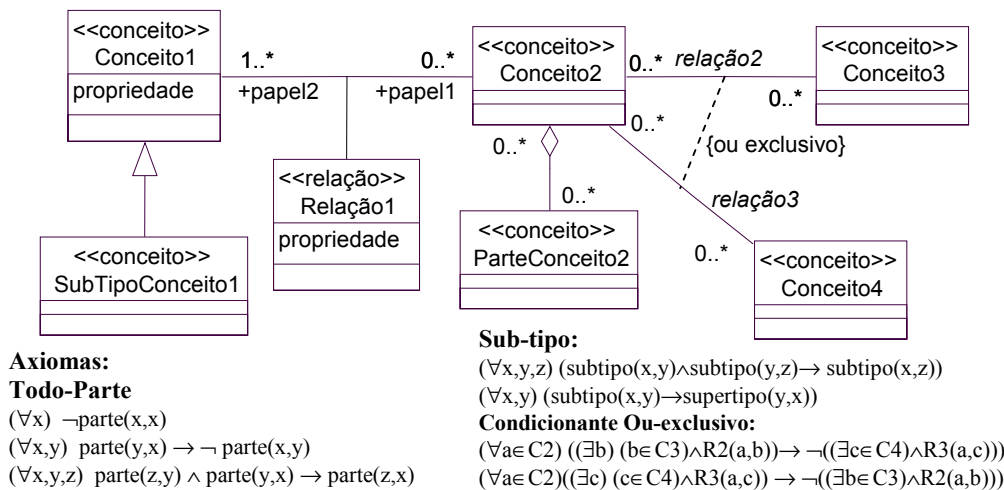


Figura 1 – Perfil UML para representar ontologias e axiomas associados.

### 3. Uma Ontologia de Qualidade de Software

De acordo com o método SABiO, primeiro identificou-se o propósito da ontologia. Quer se falar de qualidade de software de forma ampla, permitindo tratar uniformemente produtos e processos de software. Tem-se como principal uso pretendido para a ontologia a sua utilização como uma especificação conceitual para o desenvolvimento e integração de ferramentas de apoio aos processos de Garantia da Qualidade e Avaliação e Melhoria de Processos. Contudo, devido a limitações de espaço, o uso efetivo da ontologia proposta para esse fim não é aqui discutido.

Uma vez que processos de software são uma parte importante do universo de discurso da ontologia de qualidade de software, e intrinsecamente relacionados ao propósito estabelecido, a ontologia aqui proposta foi desenvolvida de forma integrada com a ontologia de processo de software proposta em [Falbo e Bertollo, 2005].

Como o escopo da ontologia é bastante complexo, optou-se por desenvolver três sub-ontologias integradas, tratando, respectivamente, da estrutura de modelos de

qualidade, medição e avaliação. A seguir, as duas primeiras são apresentadas. Novamente devido a limitações de espaço, a sub-ontologia de avaliação não é aqui discutida. A ontologia completa encontra-se descrita em [Dal Moro 2008], incluindo dicionário de termos e detalhes de sua avaliação.

A integração com a ontologia de processo de software é mostrada nos modelos conceituais apresentados por meio de uma indicação antecedendo o termo (Ontologia de Processo::Termo). Para realçar, conceitos propostos em cada uma das sub-ontologias são destacados em cinza. Por fim, são apresentadas algumas instanciações da ontologia, usadas para avaliar se ela é capaz de tratar uniformemente produtos e processos.

### 3.1 – Sub-Ontologia de Modelo de Qualidade

A sub-ontologia de modelo de qualidade trata das características básicas de uma entidade (tais como processos, artefatos e recursos) que podem ser medidas, em um primeiro momento, para em seguida serem analisadas e avaliadas. Basicamente, as questões de competência que devem ser respondidas por essa sub-ontologia são:

- QC1. Qual é o tipo de uma entidade mensurável?
- QC2. Quais elementos mensuráveis caracterizam um determinado tipo de entidade mensurável?
- QC3. Quais elementos mensuráveis podem ser diretamente medidos e quais não podem?
- QC4. Como um elemento indiretamente mensurável é decomposto em outros elementos mensuráveis, de modo que possa ser medido a partir da medição destes?
- QC5. Quais elementos diretamente mensuráveis devem ser medidos para se poder medir um elemento indiretamente mensurável?

A Figura 2 mostra o modelo conceitual da sub-ontologia de modelo de qualidade, apresentando seus conceitos e relações. Um *tipo de entidade mensurável* é uma categoria de entidades mensuráveis que indica quais características (*elementos mensuráveis*) podem ser usadas para medir e avaliar a qualidade de entidades desse tipo. Exemplos desses tipos são tipos de processos, tipos de artefatos (produtos) e tipos de recursos.

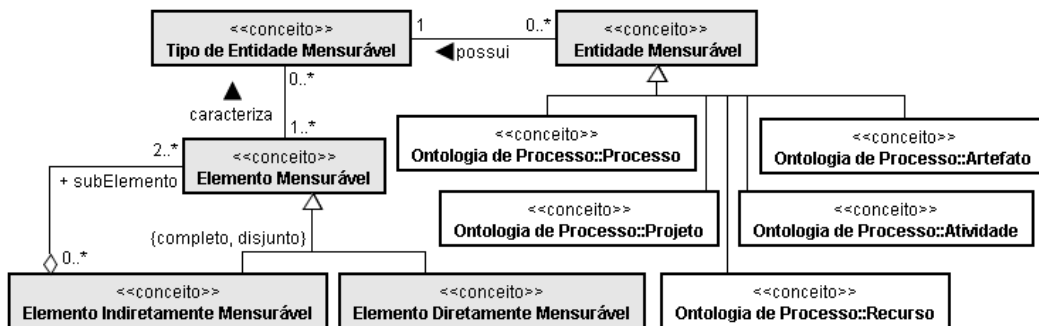


Figura 2 – O Modelo da Sub-Ontologia de Modelo de Qualidade.

Um elemento mensurável pode ser direta ou indiretamente mensurável. Um *elemento diretamente mensurável* (ou atributo) é uma propriedade física ou abstrata de um tipo de entidade mensurável que não necessita de nenhuma outra medida para ser

calculada. Já um *elemento indiretamente mensurável* é composto de outros elementos mensuráveis e, sendo assim, necessita de todas as medidas dos elementos que o compõem para ser calculado. É importante destacar que a relação de especialização entre elemento mensurável e seus subtipos representa um conjunto completo e disjunto, ou seja, todos os elementos mensuráveis são ou direta ou indiretamente mensuráveis (completo), e não há um elemento que seja direta e indiretamente mensurável (disjunto).

Uma *entidade mensurável* é um objeto (instância concreta de um tipo) que pode ser caracterizado mediante a medição dos elementos mensuráveis do seu tipo. São essas entidades que efetivamente estão sendo medidas e avaliadas. Como mostra a figura, no contexto de engenharia de software, são entidades mensuráveis, dentre outros, processos, artefatos, atividades, projetos e recursos.

No modelo apresentado, há restrições importantes que precisam ser descritas na forma de axiomas, dentre elas a seguinte:

- Um elemento indiretamente mensurável  $eim_l$  só pode ser decomposto em elementos que caracterizam o mesmo tipo de entidade mensurável  $tem_l$ .

$(\forall em_l, eim_l, tem_l) subElemento(em_l, eim_l) \rightarrow (caracteriza(em_l, tem_l) \wedge caracteriza(eim_l, tem_l))$

Vale a pena destacar que, além do axioma acima apresentado, o uso do perfil UML apresentado na Figura 1 indica outros axiomas, tal como a transitividade na relação subelemento. Esses axiomas, contudo, não são aqui apresentados.

Por fim, a Tabela 1 apresenta dois exemplos de instanciação da sub-ontologia de modelo da qualidade, na qual a primeira coluna representa um exemplo relacionado a processo de software e a segunda um exemplo relacionado a produto de software.

**Tabela 1 – Instanciação da Sub-Ontologia de Modelo de Qualidade.**

	<b>Processo</b>	<b>Produto</b>
<b>Tipo de Entidade Mensurável</b>	Processos de uma Unidade Organizacional	Especificação de Requisitos
<b>Entidade Mensurável</b>	Processo de Gerência de Projetos da Divisão de Desenvolvimento de Sistemas da Empresa X	Especificação de Requisitos do Projeto 1
<b>Elemento Mensurável</b>	Resultado Esperado do MPS.BR GPR 3 <sup>1</sup> em uma Unidade Organizacional	Notação da Especificação <sup>2</sup>

### 3.2 – Sub-Ontologia de Medição

A sub-ontologia de medição trata, como o próprio nome sugere, da medição de entidades. As questões de competência a serem respondidas por essa sub-ontologia são:

- QC1. Que entidade está sendo medida em uma medição?  
 QC2. Qual é o tipo da entidade que está sendo medida em uma medição?  
 QC3. Que elementos de um tipo de entidade podem ser medidos para se avaliar a qualidade de uma entidade associada a esse tipo?  
 QC4. Qual é o elemento da entidade que está sendo medido em uma medição?  
 QC5. Que medidas podem ser utilizadas para quantificar um elemento?

<sup>1</sup> GPR 3 é um resultado esperado do Processo de Gerência de Projetos do MPS.BR, que diz que o modelo e as fases do ciclo de vida do projeto devem ser definidos

<sup>2</sup> Refere-se à notação, incluindo formatação e padrão de nomes, empregada na especificação.



- QC6. Que medidas estão sendo aplicadas em uma medição de um elemento?
- QC7. Quanto à dependência de uma medida em relação a outras, qual a natureza de uma medida?
- QC8. Que medidas precisam ser medidas para que uma medida derivada possa ser computada?
- QC9. Qual a unidade de medida que expressa uma determinada medida?
- QC10. Qual é a escala de uma medida?
- QC11. Qual é o tipo de uma escala?
- QC12. Quais são os valores de uma escala?
- QC13. Qual o valor medido em uma medição?
- QC14. Que procedimentos de medição podem ser usados para calcular uma medida?
- QC15. Que procedimentos de medição estão sendo adotados em uma medição?

A Figura 3 mostra o modelo conceitual da sub-ontologia de medição, destacando em cinza os conceitos dessa sub-ontologia.

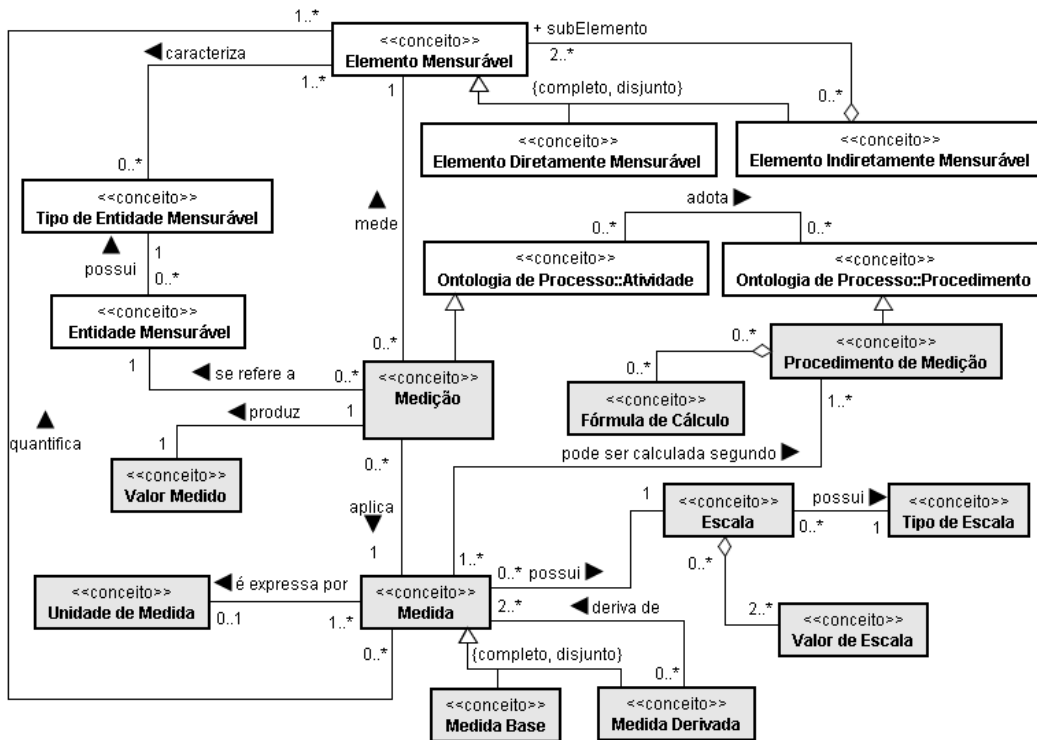


Figura 3 – O Modelo da Sub-Ontologia de Medição.

A *medição* é uma atividade que tem por objetivo medir uma característica (elemento mensurável) de uma entidade, aplicando uma *medida* para obter um *valor*. Uma medida pode ser *base*, caso ela seja funcionalmente independente de outras medidas, ou *derivada*, caso seja definida como função de duas ou mais medidas [ISO/IEC 2002].

Uma medida possui uma escala relacionada e pode ser expressa segundo uma unidade de medida. Além disso, uma medida é calculada segundo um procedimento de medição que pode conter uma fórmula de cálculo. Ao se aplicar uma medida em uma

medição, o procedimento de medição adotado deve ser apropriado para a medida em questão. Vale ressaltar que nem toda medida é expressa por alguma unidade de medida.

No modelo apresentado na Figura 3, há diversas restrições importantes que não são capturadas e, portanto, foram descritas na forma de axiomas, dentre elas:

- Uma medida base quantifica elementos diretamente mensuráveis, enquanto uma medida derivada quantifica elementos indiretamente mensuráveis.

$$(\forall em1, m1) \text{quantifica}(m1, em1) \wedge \text{elementoDiretamenteMensuravel}(em1) \rightarrow \text{medidaBase}(m1)$$

$$(\forall em1, m1) \text{quantifica}(m1, em1) \wedge \text{elementoIndiretamenteMensuravel}(em1) \rightarrow \text{medidaDerivada}(m1)$$

- Uma medida  $m_1$  só pode ser aplicada em uma medição  $med_1$ , se a medição  $med_1$  estiver medindo um elemento mensurável  $em_1$  quantificável pela medida  $m_1$ .

$$(\forall em1, m1, med1) \text{aplica}(med1, m1) \rightarrow (\text{mede}(med1, em1) \wedge \text{quantifica}(m1, em1))$$

- Uma medição  $med_1$  que se refere a uma entidade mensurável  $entm_1$  só pode medir elementos mensuráveis que caracterizam o tipo dessa entidade.

$$(\forall em1, tem1, entm1, med1) (\text{seRefereA}(med1, entm1) \wedge \text{mede}(med1, em1)) \rightarrow (\text{possui}(entm1, tem1) \wedge \text{caracteriza}(em1, tem1))$$

- Uma atividade de medição  $med_1$  que aplica uma medida  $m_1$  só pode adotar um procedimento de medição  $pm_1$  se a medida  $m_1$  puder ser calculada segundo esse procedimento de medição.

$$(\forall m1, med1, pm1) (\text{aplica}(med1, m1) \wedge \text{adota}(med1, pm1)) \rightarrow \text{podeSerCalculadaSegundo}(m1, pm1)$$

- Se uma medição  $med_1$  aplica uma medida  $m_1$  para medir um elemento diretamente mensurável  $em_1$ , então a medida  $m_1$  deve ser uma medida base que quantifique o elemento diretamente mensurável  $em_1$ .

$$(\forall med1, m1, em1) (\text{mede}(med1, em1) \wedge \text{elementoDirMensuravel}(em1) \wedge \text{aplica}(med1, m1)) \rightarrow (\text{quantifica}(m1, em1) \wedge \text{medidaBase}(m1))$$

- Se uma medição  $med_1$  aplica uma medida  $m_1$  para medir um elemento indiretamente mensurável  $em_1$ , então a medida  $m_1$  deve ser uma medida derivada que quantifique esse elemento.

$$(\forall med1, m1, em1) (\text{mede}(med1, em1) \wedge \text{elementoIndMensuravel}(em1) \wedge \text{aplica}(med1, m1)) \rightarrow (\text{quantifica}(m1, em1) \wedge \text{medidaDerivada}(m1))$$

- Se uma medida derivada  $md_1$  quantifica um elemento indiretamente mensurável  $eim_1$  e ela deriva de uma medida  $m_2$ , então existe um elemento mensurável  $em_2$  que é quantificado pela medida  $m_2$  e é um sub-elemento de  $eim_1$ .

$$(\forall md1, m2, eim1, em2) (\text{quantifica}(md1, eim1) \wedge \text{derivaDe}(md1, m2)) \rightarrow (\exists em2 (\text{quantifica}(m2, em2) \wedge \text{subElemento}(em2, eim1))$$

Finalmente, a Tabela 2 apresenta dois exemplos de instanciação de alguns dos conceitos da sub-ontologia de medição, tomando por base os mesmos exemplos usados para exemplificar a instanciação da sub-ontologia de modelo de qualidade, ou seja, a primeira coluna trata da medição de um processo de gerência de projetos de uma unidade organizacional (usando como base o MPS.BR [Softex, 2007]), enquanto a segunda trata dos aspectos relacionados à medição de uma especificação de requisitos (tomando por base [Beaufound et al., 1997]).

Tabela 2 – Instanciação da Sub-Ontologia de Medição.

	Processo	Produto
<b>Medida</b>	Grau de Implementação de Resultado Esperado na Unidade Organizacional	Correção da Notação
<b>Escala</b>	Escala de Caracterização Agregada do Grau de Implementação de um Resultado Esperado para a Unidade Organizacional	Escala para Caracterização da Correção da Notação de uma Especificação de Requisitos
<b>Tipo de Escala</b>	Nominal	Nominal
<b>Valores de Escala</b>	[T, L, P, N, NA, F] <sup>3</sup>	Sim/Não
<b>Procedimento de Medição</b>	<ul style="list-style-type: none"> <li>• Todos X (i.e., todos T ou todos L ou todos P ou todos N) → X</li> <li>• Todos projetos terminados X (i.e., todos T ou todos L ou todos P ou todos N) e os incompletos NA (Não Avaliado) → X</li> <li>• Todos T ou L → L</li> <li>• Todos T ou L e os incompletos NA (Não Avaliado) → L</li> <li>• Existe P, mas não existe N (Pode existir NA - Não Avaliado) → L ou P</li> <li>• Existe N → N, P ou L</li> <li>• Fora do Escopo → F</li> </ul>	A especificação está escrita utilizando de forma correta a notação pré-definida no método de especificação?
<b>Medição</b>	Caracterizar o grau de implementação do resultado esperado de processo MPS.BR GPR3 na Unidade Organizacional UO	Avaliar a correção da notação da Especificação do Projeto 1
<b>Valor Medido</b>	Largamente Implementado.	Sim

### 3.3 – Comparação com a Ontologia de Medição Proposta em [Bertoa et al. 2006]

Dentre os trabalhos citados, a ontologia de medição proposta por Bertoa et al. (2006) é a que mais se aproxima à ontologia de qualidade aqui apresentada, por ter um caráter mais geral. De fato, algumas partes de sua conceituação, tal como a que trata de escalas, tipos e valores de escala, foram reutilizadas. Contudo, há diferenças significativas, conforme discutido a seguir.

Em linhas gerais, nossa sub-ontologia de Modelo de Qualidade corresponde à sub-ontologia de Caracterização e Objetivos da Medição de Software de Bertoa et al. A percepção de que há entidades e tipos de entidades está presente em ambas as ontologias. Entretanto, há diversas diferenças. Bertoa et al. indicam que uma entidade pode ser composta de outras entidades. Isso é válido tanto para processos quanto para produtos. Entretanto, uma vez que um recurso (por exemplo, humano) também pode ser uma entidade mensurável no contexto de software (por exemplo, avaliação de produtividade), optamos por não incluir essa relação. Outro ponto comum é que ambas tratam de elementos que caracterizam os tipos de entidades. Bertoa et al. utilizam o termo “atributo” para designar o que chamamos de “elemento mensurável”. Preferimos usar um termo mais geral, pois em avaliações de processo avaliam-se tanto atributos quanto resultados de processo e, portanto, o termo “atributo” pode provocar confusão. Além disso, neste trabalho fazemos distinção entre elementos direta e indiretamente

<sup>3</sup> Os valores de escala usados na avaliação de processos do MPS.BR são: Fora de Escopo (F), Não Avaliado (NA), Totalmente (T), Largamente (L), Parcialmente (P) e Não Implementado (N). Ainda que os quatro últimos valores tenham uma ordem, os dois primeiros não têm, razão pela qual essa escala é considerada nominal.

mensuráveis, capturando o fato de elementos indiretamente mensuráveis serem compostos de sub-elementos. Por fim, Bertoa et al. consideram os conceitos de Modelo de Qualidade, Necessidade de Informação e Conceito Mensurável que julgamos não serem consensuais na área e que, portanto, não deveriam fazer parte da ontologia. Sobretudo o último é muito questionável, dado que sua definição é “um relacionamento abstrato entre atributos de entidades e necessidades de informação”.

Em relação à sub-ontologia de Medição, que corresponde às demais sub-ontologias de [Bertoa et al., 2006], há também algumas diferenças: (i) preferimos não considerar indicador como um tipo de medida como em [Bertoa et al., 2006], pois consideramos que ser um indicador é um papel de uma medida e não um tipo de medida; (ii) uma vez que fazemos uma distinção entre elementos direta e indiretamente mensuráveis, optamos por relacionar medidas base e derivadas diretamente com esses dois conceitos, respectivamente; (iii) tendo em vista que a ontologia aqui proposta foi construída de forma integrada com a ontologia de processo de software, a parte que trata de abordagens de medição em [Bertoa et al. 2006] é bastante diferente.

Por fim, em [Bertoa et al. 2006], não se discutem restrições e sua formalização. Neste trabalho, essa foi uma forte preocupação, bem como a de explicitamente avaliar se a conceituação proposta aplica-se tanto a produtos quanto a processos de software.

#### 4. Conclusões

Qualidade de software envolve diversas perspectivas, com destaque para os enfoques centrados em processos e produtos de software. Neste trabalho apresentamos uma ontologia de qualidade de software, que procura conciliar essas duas perspectivas, abrindo espaço para uma visão parcialmente unificada.

Em continuação a este trabalho, estamos desenvolvendo ferramentas de apoio aos processos de Garantia da Qualidade e Avaliação e Melhoria de Processos de Software para um ambiente de desenvolvimento de software, usando essa ontologia como uma especificação base para o desenvolvimento e integração de ferramentas. Resultados preliminares apontam para a grande utilidade da ontologia, uma vez que está sendo possível não somente integrar dados das ferramentas, mas também funcionalidades providas para diferentes propósitos, sobretudo no que se refere à sua aplicação para a avaliação da qualidade de produtos e processos.

#### Referências

- Beaufound, C.E.C., Werner, C.A.L., Rocha, A.R.C. (1997) Manual para Controle da Qualidade de Especificações Orientadas a Objetos, Relatório Técnico ES-428/97, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ.
- Bertoa, M.F., Vallecillo, A., García, F. (2006) “An Ontology for Software Measurement”, In: *Ontologies for Software Engineering and Software Technology*, Springer-Verlag Berlin Heidelberg.
- Dal Moro, R. (2008) Avaliação e Melhoria de Processos de Software: Conceituação e Definição de um Processo para Apoiar a sua Automatização, Dissertação de Mestrado, Programa de Pós-Graduação em Informática - UFES.

- Duarte, K.C., Falbo, R.A. (2000) “Uma Ontologia de Qualidade de Software”, Anais do VII Workshop de Qualidade de Software, XIV Simpósio Brasileiro de Engenharia de Software, pp. 275-285, João Pessoa, Brasil.
- Falbo, R. A. (2004) “Experiences in Using a Method for Building Domain Ontologies” Proc. of the 16th International Conference on Software Engineering and Knowledge Engineering, International Workshop on Ontology In Action, Banff, Canada.
- Falbo, R.A., Bertollo, G. (2005) “Establishing a Common Vocabulary for Software Organizations Understand Software Processes”, International Workshop on Vocabularies, Ontologies and Rules for the Enterprise, Enschede, The Netherlands.
- Fuggetta, A. (2000) “Software Process: A Roadmap”, In: Proceedings of The Future of Software Engineering, ICSE’2000, Limerick, Ireland.
- Guizzardi, G. (2005) *Ontological Foundations for Structural Conceptual Models*, Universal Press, The Netherlands.
- ISO/IEC (1999) ISO/IEC 14598 *Information Technology – Software product evaluation*.
- ISO/IEC (2001), ISO/IEC 9126-1:2001 – *Software Engineering – Product Quality – Part 1: Quality model*.
- ISO/IEC (2002) ISO/IEC 15939:2002 – *Software Engineering – Software Measurement Process*.
- ISO/IEC (2003), ISO/IEC 15504 - *Information Technology – Process Assessment*.
- ISO/IEC (2008), ISO/IEC 12207 – *System and Software Engineering - Software Life Cycle Processes, 2<sup>nd</sup> edition*.
- Jasper, R., Uschold, M. (1999) “A Framework for Understanding and Classifying Ontology Applications”, Proceedings of the IJCAI99 Workshop on Ontologies and Problem-Solving Methods, Stockholm, Sweden.
- Mian, P.G., Falbo, R.A. (2003) “Supporting Ontology Development with ODED”, Journal of the Brazilian Computer Science, vol. 9, no. 2, pp 57-76.
- Martin, M. A., Olsina, L. (2003) "Towards an Ontology for Software Metrics and Indicators as the Foundation for a Cataloging Web System", First Latin American Web Congress (LA-WEB'03).
- Rocha A.R.C., Maldonado, J.C., Weber, K.C. (2001) *Qualidade de Software: Teoria e Prática*, Prentice Hall, São Paulo.
- SEI (2006), CMMI for Development – v1.2, Pittsburgh: Software Engineering Institute.
- Softex (2007) MPS.BR – Melhoria de Processo do Software Brasileiro: Guia Geral, Versão 1.2.