

# Plano de Garantia da Qualidade na Perspectiva do Desenvolvimento Distribuído de Software

Fábio Levy Siqueira, Paulo Sérgio Muniz Silva

Departamento de Engenharia de Computação e Sistemas Digitais – Escola Politécnica  
da Universidade de São Paulo  
Av. Prof. Luciano Gualberto, trav.3, nº 158 – 05508-900 – São Paulo – SP – Brazil  
{levy.siqueira, paulo.muniz}@poli.usp.br

**Abstract.** *The quality assurance plan defines the philosophy and activities to be executed in a project by the quality assurance staff. To create this document it is necessary to consider the development process and the project characteristics, and observe the available limitations and possibilities. A distributed software development requires consideration of another group of characteristics related to the particularities of this type of development. This paper analyzes how the software quality assurance plan is influenced by the distributed software development, and discusses some fundamental issues that need to be considered by the quality assurance staff.*

**Resumo.** *O plano de garantia da qualidade define a filosofia e as atividades que serão executadas dentro de um projeto pela equipe de garantia da qualidade. Para criar esse documento é preciso considerar as características do processo de desenvolvimento e do projeto, observando as limitações e as possibilidades disponíveis. No caso de um desenvolvimento distribuído de software existe um outro conjunto de características que precisa ser considerado, referente às particularidades dessa forma de desenvolvimento. Este artigo analisa como o plano de qualidade é afetado pelo desenvolvimento distribuído, evidenciando algumas questões fundamentais que precisam ser consideradas pela equipe de garantia da qualidade.*

## 1. Introdução

A complexidade dos softwares exigidos pelo mercado torna praticamente inviável o desenvolvimento realizado por uma única pessoa. Mesmo quando o objetivo é criar sistemas de pequeno porte e com lógica de negócio simples existem outros fatores como a qualidade e o prazo que podem exigir mais de uma pessoa para resolver o problema. Com isso, são necessárias equipes de desenvolvimento de software que, além do imprescindível conhecimento técnico, tenham habilidades gerenciais e de comunicação para dividir o esforço de forma colaborativa.

É de se esperar que as partes envolvidas em um projeto trabalhem fisicamente próximas para permitir, principalmente, um maior e mais eficiente fluxo de informações (Cockburn, 2002). No entanto em algumas situações esse formato de solução pode não ser o mais viável. A complexidade de um projeto pode obrigar que ele seja realizado por diversas empresas espalhadas por uma mesma cidade; a busca por um especialista, seja ele uma pessoa ou uma organização, pode levar a outros estados; a necessidade competitiva de cortar custos ao utilizar mão-de-obra barata e ainda assim qualificada

pode envolver organizações em outros países. Nesses casos em que as pessoas envolvidas no projeto não podem trabalhar em proximidade física é necessário realizar uma outra forma de desenvolvimento: o desenvolvimento distribuído de software.

Apesar de o desenvolvimento distribuído de software ser uma solução cada vez mais comum, ainda existem dúvidas de como adequar os processos para essa forma de desenvolvimento. A separação espacial e temporal das partes envolvidas causa diversos problemas e dificuldades no desenvolvimento de software, como, por exemplo, na engenharia de requisitos (Damian e Zowghi, 2003), na gerência de riscos (Prikladnicki e Yamaguti, 2004), na gerência de configuração (Karolak, 1998) e na garantia da qualidade.

Considerando isso, pretende-se explorar nesse artigo como o desenvolvimento distribuído de software afeta a garantia da qualidade. Para isso, será abordada uma atividade essencial para a sua realização: a criação do plano de garantia da qualidade. Esse plano expõe a estratégia da garantia da qualidade dentro de um projeto (Schulmeyer e McManus, 1998), definindo a filosofia e as atividades a serem realizadas. Pretende-se, assim, explorar um ponto chave à garantia da qualidade na perspectiva do desenvolvimento distribuído de software, imprescindível mesmo durante a implementação de um arcabouço de melhoria de processo (como, por exemplo, o CMMI (CMMI Product Team, 2002) ou o ISO/IEC 15504 (ISO, 1998)). Como base para a discussão será considerado um padrão para a criação desse plano, o IEEE 730 (IEEE, 2002). Esse padrão define um conjunto de seções que um plano deve conter, apontando as atividades que precisam ser descritas e também um conjunto mínimo requerido de documentos a serem criados e revisões a serem realizadas.

Dessa forma, o objetivo principal do artigo é fornecer um quadro geral das questões fundamentais que a gerência de qualidade precisa considerar para a criação de um plano de qualidade, evidenciando a complexidade da questão. Para isso, esse trabalho está estruturado da seguinte maneira: as seções 2 e 3 apresentam, respectivamente, o desenvolvimento distribuído de software e a garantia da qualidade; e a seção 4 discute alguns trabalhos relacionados. Em seguida, na seção 5, é analisado como as atividades definidas pelo padrão de criação do plano de qualidade IEEE 730 são afetadas por essa forma de desenvolvimento. Finalmente, a seção 6 apresenta a conclusão e os trabalhos futuros, destacando os objetivos de um estudo de caso a ser realizado.

## **2. O Desenvolvimento Distribuído de Software (DDS)**

De forma geral, o desenvolvimento distribuído de software é aquele realizado por pessoas que não estão dispostas em proximidade física. Essa dispersão espacial, e eventualmente temporal, traz diversas vantagens ao desenvolvimento de software, o que faz com que o DDS seja uma opção bastante interessante ou até uma necessidade em alguns casos. Talvez a principal vantagem de se realizar o DDS seja aproveitar a mão-de-obra mais barata e mesmo assim especializada disponível em outros países (Carmel, 1999). Mas além disso, existem outras motivações como, por exemplo, a busca por uma mão-de-obra especializada, a redução do prazo de entrega ao realizar o desenvolvimento em 24 horas, o custo alto da viagem e da transferência de funcionários, a necessidade de haver pessoas da equipe próximas ao cliente e a necessidade de dividir a equipe entre

locais e organizações devido ao tamanho do projeto ou à falta de espaço (Carmel, 1999); (Herbsleb e Moitra, 2001); (Layzell et al., 2000).

Da mesma forma que existem diversas motivações, existem também diversos problemas ao realizar um desenvolvimento desta forma. Ao observar a experiência prática sobre esse assunto, percebe-se que a separação das pessoas ocasiona diversas dificuldades que podem até inviabilizar o desenvolvimento dessa forma. Entre os problemas mais evidentes podem ser citados: diferenças de língua e cultura, incompatibilidade de ferramentas e infra-estrutura, processos diferentes, dificuldade de comunicação devido à riqueza dos meios de comunicação disponíveis e às diferenças de fusos horários (Battin et al., 2001); (Carmel, 1999); (Karolak, 1998). Mas, além desses problemas, existem inúmeros outros como, por exemplo, ciúmes de engenheiros forçados a treinar funcionários mais baratos, legislações diferentes, falta de senso de equipe, falta de confiança no grupo (Battin et al., 2001); (Carmel, 1999); (Ebert e De Neve, 2001).

Considerando os problemas existentes, o DDS não pode ser aplicado sem um planejamento adequado para que se consiga gerenciar todos os problemas, riscos e dificuldades, tentando contornar os problemas e se beneficiar das vantagens disponíveis.

### **3. Garantia da Qualidade**

A busca por qualidade é uma necessidade para qualquer organização que desenvolve software, assim como a busca por preços mais baixos e prazos mais curtos. Os clientes e usuários exigem produtos que consigam atender os requisitos definidos e também suas expectativas, sejam elas explícitas ou implícitas.

Para obter um produto de qualidade, a organização precisa pôr em foco alguns fatores que influenciam diretamente o resultado obtido: a qualidade do pessoal, o custo, o tempo e o calendário, a tecnologia de desenvolvimento e a qualidade processo (Sommerville, 2001). Dessa forma, o processo é um dos pontos importantes para a obtenção de qualidade no produto, uma vez que garante a verificação e a validação durante a produção e permite a criação do produto certo (Unhelkar, 2003). Com isso, é importante concentrar-se na obtenção de um processo de qualidade, seja ao implantar arcabouços como o CMMI (CMMI Product Team, 2002) e a norma ISO/IEC 15504 (ISO, 1998) para a melhoria do processo, ou simplesmente ao executar atividades que permitam tornar o processo mais adequado à criação do produto idealizado. E, ao aumentar a qualidade de um processo, é esperado que o produto em si tenha uma melhoria em sua qualidade.

Nesse sentido, as atividades de garantia da qualidade têm um papel importante na melhoria da qualidade do processo ao definir os padrões que devem ser utilizados para verificar a adequação do processo e alterá-lo se necessário. De forma geral, a garantia da qualidade é o conjunto de atividades que demonstram a capacidade do processo em gerar um produto com qualidade (Schulmeyer e McManus, 1998). Para realizar isso, é necessário criar um grupo que seja independente e consiga realizar com liberdade as atividades relacionadas à gerência da qualidade. A primeira atividade desse grupo dentro de um projeto é criar um plano de garantia da qualidade do software que definirá os padrões e a filosofia principal das questões relativas à qualidade. Mas além disso, esse grupo deve executar outras atividades, como, por exemplo, desenvolver

ferramentas para auxiliar a tarefa de garantia da qualidade, manter a gerência ciente do status da qualidade em projetos de desenvolvimento de software e participar de revisões de projeto do software, testes, controle de configuração, aviso e solução de problemas e controle de mudanças (Schulmeyer e McManus, 1998).

#### **4. Trabalhos Relacionados**

Em um desenvolvimento distribuído de software podem-se ter grandes diferenças entre as organizações, seja no processo de desenvolvimento empregado, na preocupação com a qualidade ou até na cultura organizacional de um modo geral. Tratar esses problemas e chegar a um acordo de um processo com qualidade que seja adequado pode ser uma tarefa complicada, ainda mais se forem consideradas outras variáveis presentes no desenvolvimento distribuído de software, como a cultura, a língua, etc.

Pensando nesses problemas, Kobitzsch et al. (2001) sugerem alguns modelos de cooperação entre organizações e observam como a qualidade deve ser tratada. Em empresas legalmente associadas é preciso chegar-se a um consenso do que é qualidade, enquanto que em empresas contratadas pode-se selecionar uma que apresente maior adequação, por exemplo através de certificações. Enquanto que no último caso trata-se de uma mera questão de escolha, o primeiro pode exigir treinamento intenso para a obtenção do nível desejado de qualidade. Independentemente da relação legal entre as organizações, o importante é que seja possível obter um entendimento comum em relação à qualidade, para posteriormente se montar um modelo de qualidade adequado.

Uma outra solução para essa questão é a utilização de um processo de desenvolvimento moldado para o desenvolvimento distribuído de software e que considere adequadamente a qualidade. Maidantchik e Da Rocha (2002) propõem a adaptação do padrão ISO/IEC 12207 (ABNT, 1998) para o desenvolvimento distribuído de software, montando um arcabouço que segue o nível de maturidade da organização – avaliado segundo os critérios do CMM (Paulk et al., 1993).

Focalizando mais exclusivamente nas atividades de garantia da qualidade, existem algumas propostas que se preocupam com sua execução adequada em ambientes distribuídos. Em uma delas, Bianchi et al. (2001) propõem um modelo GQM (*goal-question-metric*) distribuído que considera as características de cada organização para montar um modelo GQM mais adequado ao DDS, por exemplo, ao direcionar um conjunto de objetivos para determinadas organizações ou criar métricas diferentes para uma mesma questão em diferentes organizações. Existem também alguns trabalhos que enfocam, por exemplo, em como realizar inspeções de maneira distribuída (Caivano et al., 2001); (Hedberg e Harjumaa, 2002) e também em algumas métricas que podem ser aplicadas (Bourgault et al., 2002); (Dutoit, 1998).

#### **5. O DDS e o Plano de Garantia da Qualidade do Software**

A existência de diversas organizações envolvidas no desenvolvimento de software pode fazer com que a organização contratante exija das demais a criação de um plano de qualidade que satisfaça a alguns requisitos determinados. No caso de *outsourcing* (atividades de engenharia de software contratadas de um terceiro (Pressman, 2001)), os planos de qualidade tratarão pouco sobre as questões da distribuição geográfica, já que cada uma das organizações será relativamente independente em seu desenvolvimento.

Nos outros casos de desenvolvimento distribuído – em que as organizações não são contratadas (matriz-filial, *joint venture* ou duas filiais, por exemplo), as equipes são relativamente dependentes ou até quando algumas organizações são imaturas nas questões da qualidade –, não é possível criar um plano de garantia da qualidade do software (PGQS) sem pensar nas características do DDS. Dependendo do caso, pode ser necessário criar um único plano de qualidade que considere todas as organizações.

Nessas situações em que é preciso considerar as particularidades do DDS, o primeiro passo é a conscientização da importância da garantia da qualidade entre as diversas organizações, caso ela ainda não faça parte da cultura organizacional, e posteriormente a definição de um entendimento comum das questões de qualidade (Kobitzsch et al., 2001). Uma vez atingido o consenso sobre a questão, pode ser ainda necessário montar uma equipe de garantia da qualidade do software, a qual pode até ser distribuída.

Considerando a existência de uma equipe de garantia da qualidade, a criação do PGQS deve ser a sua primeira atividade executada dentro de um determinado projeto. Esse documento deverá descrever os planos e atividades realizadas pela equipe de garantia da qualidade (IEEE, 1995), definindo a filosofia de qualidade adotada para o determinado projeto. Para observar os principais problemas que afligem a criação de um PGQS para um desenvolvimento distribuído, será utilizado como exemplo de conteúdo o que é definido pelo padrão IEEE 730 (IEEE, 2002). Esse padrão especifica um conjunto de seções que precisam ser detalhadas para a criação de um plano de qualidade, abordando objetivamente o conteúdo de cada uma delas. Devido à concisão do padrão, também foi utilizado para a análise das questões o padrão IEEE 730.1 (IEEE, 1995) que, apesar de ser relativo à versão de 1989 do padrão IEEE 730, oferece uma explicação mais aprofundada da filosofia e do conteúdo de cada uma das seções do padrão.

Seguindo o padrão, analisa-se a seguir como o DDS afeta cada uma das seções, utilizando para isso o conhecimento obtido pela análise de diversos relatos de experiência na literatura do DDS. Devido à complexidade do DDS e da garantia da qualidade, serão apenas abordados alguns problemas principais, havendo assim diversas outras questões que também precisam ser consideradas.

### **5.1. Gerência**

Essa seção do plano de qualidade tem como objetivo definir a estrutura organizacional, as tarefas, os papéis e as responsabilidades envolvidas na garantia da qualidade. Para isso, precisam ser consideradas algumas particularidades do DDS, como as diferenças culturais que podem fazer com que as pessoas tenham problemas ao se reportar a alguém com um cargo inferior de uma outra organização (Haywood, 2000), mas encarregado de uma determinada atividade de garantia da qualidade. Além disso, a existência de diversas organizações faz com que a identificação das pessoas envolvidas nas atividades de garantia da qualidade seja mais trabalhosa, exigindo o conhecimento da estrutura das demais organizações.

Alguns outros problemas gerenciais também precisam ser considerados, como agendar reuniões de decisão gerencial e distribuir os documentos previstos pelo PGQS. No caso das reuniões de decisão gerencial, é preciso considerar principalmente a

diferença de tempo entre os locais e o calendário (feriados locais e religiosos, férias dos funcionários e outras restrições). Para a distribuição dos documentos previstos pelo PQGS, a dispersão das pessoas em diversas localidades e em fusos horários diferentes exige um maior esforço de planejamento para que a entrega seja feita para a pessoa certa e no momento certo. Além disso, a entrega da versão correta pode ser dificultada pela existência de diferentes versões entre as localidades – sendo necessária uma gerência de configuração adequada.

## **5.2. Documentação**

A grande dificuldade para a definição da documentação necessária para o projeto é a diversidade das culturas organizacionais. Cada organização pode ter um conjunto próprio de documentos que podem até possuir o mesmo conteúdo, mas que estão organizados de forma diferente. Também pode haver necessidade burocrática de um determinado padrão, ou outros pequenos detalhes que podem tornar difícil a criação de um conjunto único de documentos para todas as organizações envolvidas no projeto e seus respectivos critérios para a auditoria e inspeção.

Uma outra fonte de dificuldade é a diferença de línguas, que obriga a existência de uma língua única em todos os documentos que serão usados entre diversas pessoas. A falta de proficiência na língua pode tornar a criação da documentação demorada e também gerar problemas sintáticos e semânticos sérios no texto, dificultando o entendimento da documentação. Além disso, as diferenças culturais podem trazer outros problemas, como mudanças no conteúdo do texto, fazendo-o, por exemplo, mais conciso e burocrático – o que talvez dificulte a definição do critério de adequação do documento.

## **5.3. Padrões, Práticas, Convenções e Métricas**

A definição de padrões, práticas e convenções é algo de suma importância no caso de um DDS. A existência de diversas organizações – cada uma com sua cultura organizacional e infra-estruturas diferentes – pode fazer com que exista uma enorme variedade de padrões sendo seguidos, práticas aplicadas e convenções utilizadas, o que gera sérios problemas na comunicação e na colaboração entre as pessoas. Dependendo da relação legal entre as organizações, pode ser trabalhoso chegar a um consenso nesse assunto, já que cada organização defende o que ela utiliza. Além disso, qualquer mudança gerará uma grande dificuldade para a adoção de qualquer outro padrão, prática ou convenção, necessitando um treinamento adequado.

Um outro ponto importante nesse assunto é que há a necessidade de serem considerados alguns padrões, práticas e convenções específicas devido ao DDS. Podem ser definidos, por exemplo, os meios de comunicação prioritários (Haywood, 2000), os horários adequados para se telefonar para uma outra pessoa e o padrão para a definição de horários para reuniões (importante quando não há horários de trabalho em comum). Essas definições precisam considerar aspectos sociais (possibilidade de comunicação urgente em feriados e férias) e culturais (a necessidade de haver alguém do mesmo nível hierárquico para uma conversa técnica (Haywood, 2000)), tentando diminuir os atritos entre as pessoas.

Em relação às métricas adotadas, é preciso que elas verifiquem adequadamente o processo de desenvolvimento, observando aspectos que sejam relevantes no caso do DDS, como, por exemplo, a quantidade de comunicação entre as partes (Dutoit, 1998). Uma escolha adequada dessas métricas precisa considerar alguns outros fatores do processo de desenvolvimento que se tornam importantes no DDS. Além disso, a existência de diferentes organizações pode fazer com que não seja possível utilizar as mesmas métricas em cada um dos locais. Sobre esse assunto, Bianchi et al. (2001) propõem a adaptação do paradigma GQM (*goal-question-metric*) para o DDS.

#### **5.4. Revisões de Software**

As revisões de software têm como objetivo principal avaliar a criação dos produtos de software gerados. No caso de um desenvolvimento distribuído de software é imprescindível a realização de revisões, já que há a tendência de existir diferenças de expectativas nas adequações técnicas dos produtos, além de possíveis problemas de entendimento de requisitos e uso de padrões.

No entanto, a execução dessas revisões pode ser uma atividade crítica, já que as pessoas necessárias para realizá-las podem estar distribuídas em diversas localizações. Além disso, a distância física entre elas pode tornar inviável a execução dessas reuniões de forma tradicional e as diferenças de tempo e de língua podem dificultar bastante o acerto da agenda e a execução de reuniões que sejam necessárias – mesmo pensando numa reunião virtual (por exemplo, vídeo-conferência e tele-conferência).

Dessa forma, pode ser necessário para a equipe de garantia da qualidade adaptar o processo utilizado para as revisões de software, considerando assim as características do DDS. No caso das inspeções, existem alguns trabalhos como (Caivano et al., 2001) e (Hedberg e Harjumaa, 2002) que contemplam as mudanças necessárias e sugerem algumas ferramentas que podem ser utilizadas para isso.

#### **5.5. Teste**

Os testes em um desenvolvimento distribuído de software podem ser executados de diversas formas. Dependendo da forma como o grupo está distribuído e organizado (agrupamento), pode haver uma localidade que executa todos os testes, aproveitando a diferença de tempo entre os locais para realizar os testes à noite (Herbsleb e Moitra, 2001). Com isso, é necessário que haja um planejamento adequado para a execução dos testes, considerando as possibilidades devido ao agrupamento, à diferença de tempo e às culturas organizacionais envolvidas (maior preocupação e especialidade na execução dos testes, por exemplo). A infra-estrutura também pode ser uma questão chave, se for necessária a execução dos testes em máquinas especiais ou com determinadas ferramentas.

É preciso também haver uma preocupação com a integração dos códigos criados, para permitir que a execução dos testes considere a versão mais adequada dos códigos.

#### **5.6. Aviso de Problemas e Ação Corretiva**

A definição de práticas e procedimentos adequados para o aviso de problemas e para a ação corretiva necessita de uma atenção especial devido às características do DDS. No caso do aviso de problemas, se o usuário estiver envolvido pode haver a necessidade dos

locais de atendimento serem proficientes em uma determinada língua, o que pode ser uma dificuldade especial em um atendimento centralizado. No caso específico da manutenção, além de existirem diversas formas para executá-la devido à quantidade de organizações envolvidas, a diversidade cultural pode fazer com que a importância dada a essa atividade seja maior ou menor (Karolak, 1998). A identificação dos problemas também pode ser dificultada, considerando as diferenças na infra-estrutura que podem estar envolvidas e também questões culturais que fazem com que, por exemplo, ícones e cores tenham significados semânticos diferentes – o que pode não ser visto como problema por quem faz a priorização.

Para a ação corretiva, em casos de urgência pode ser difícil encontrar uma determinada pessoa devido a diferenças de fuso horário. Mais que isso, o agrupamento e a distância física podem dificultar a identificação rápida da pessoa responsável para a correção do problema encontrado.

### **5.7. Ferramentas, Técnicas e Metodologia**

A necessidade de execução de novas atividades e as mudanças na realização de algumas já existentes faz com que sejam necessárias algumas ferramentas, técnicas e metodologias especiais para o apoio da tarefa de garantia da qualidade. A inspeção, o controle de registro e o aviso de problemas, por exemplo, podem exigir algumas ferramentas e metodologias especiais que permitam a sua realização em um ambiente distribuído. Para a escolha da ferramenta adequada é também importante notar que alguns problemas legais podem impedir a sua utilização em uma determinada localidade.

### **5.8. Controle de Mídia**

Alguns problemas provenientes das diferenças de locais, como a constante falta de energia elétrica e a fragilidade da estrutura de comunicações (Kobitzch et al., 2001), podem exigir a existência de uma infra-estrutura especial para impedir danos e facilitar a restauração de algum determinado produto de software, precisando assim ser considerada pela equipe encarregada do plano de qualidade.

### **5.9. Controle de Fornecedor**

Dependendo da relação legal e das culturas organizacionais, a definição do controle do fornecedor poderá ser um ponto crítico do PGQS. No caso de organizações contratadas que trabalhem de forma independente e que tenham uma conscientização da questão da qualidade, esta seção poderá abranger a maioria das questões do desenvolvimento distribuído (como inspecionar, verificar e validar, etc). Nos outros casos, a importância dessa seção será menor, chegando até o ponto em que não haja exatamente fornecedores quando há apenas um plano de qualidade para todas as organizações envolvidas.

### **5.10. Coleta de Registro, Manutenção e Conservação**

As diferentes culturas organizacionais podem exigir diferentes registros coletados, armazenamentos e conservações da documentação. Caso seja necessária a transferência dessas informações entre as diversas localidades, é também necessário observar a infra-



estrutura disponível, verificando se é preciso realizar essa atividade em algum horário especial para evitar uma sobrecarga nos canais de comunicação.

### **5.11. Treinamento**

As diversas mudanças necessárias nas outras seções, devidas ao DDS, provavelmente farão com que seja necessário algum treinamento especial para a execução das atividades. Além disso, as diferenças culturais podem fazer necessária a realização de palestras que permitam o entendimento das diferenças de comportamento das pessoas, ensinando como agir adequadamente (Carmel, 1999). Um outro treinamento importante pode ser o aprendizado de uma língua estrangeira, tentando permitir uma melhor comunicação e colaboração entre as partes.

### **5.12. Gerência de Risco**

Além dos riscos principais de um projeto de desenvolvimento de software tradicional, um projeto executando o DDS precisa considerar alguns outros riscos, como por exemplo: o agrupamento e a distância física podem gerar problemas de falta de confiança e senso de equipe, a diferença de tempo pode impedir a comunicação entre algumas pessoas, as diferenças nas línguas podem ocasionar problemas de comunicação e colaboração, um problema na infra-estrutura de uma organização pode causar problemas de quebra da comunicação e pode ser difícil a execução de uma determinada atividade conforme o planejado, devido a uma determinada cultura organizacional.

Dessa forma, é imprescindível identificar os novos riscos, avaliar seus diferentes impactos e identificar estratégias adequadas para a gestão deles. As particularidades do DDS também podem dificultar o monitoramento dos riscos devido à distância física, a qual pode impedir que o gerente realize uma observação detalhada do andamento do projeto.

Finalmente, deve-se considerar que, caso seja criado um único plano de qualidade para todas as organizações, a própria criação do plano sofre as conseqüências do DDS, como por exemplo:

- o agrupamento da equipe pode fazer com que nem todas as pessoas envolvidas nessa atividade estejam em um mesmo local físico;
- a distância física pode inviabilizar (financeiramente) a reunião dessas pessoas;
- as diferenças de tempo podem dificultar o agendamento de uma reunião distribuída;
- as diferenças culturais podem atrapalhar consideravelmente o progresso da reunião, devido a atritos, mal entendidos e comportamentos diferentes;
- as diferenças de línguas podem dificultar o entendimento dos assuntos discutidos e das idéias colocadas;
- as diferenças dos locais podem dificultar a agenda e a execução da reunião;
- as infra-estruturas de algumas organizações podem não permitir a execução adequada de reuniões virtuais;
- as diferentes culturas organizacionais podem fazer com que a atividade seja conduzida de uma forma diferente do que algumas pessoas estão acostumadas;

- a relação legal pode fazer com que algumas pessoas se sintam menos importantes para a definição e o planejamento.

Com isso, a criação de um PGQS precisa ser devidamente planejada e organizada para tentar diminuir os possíveis problemas que ocorrerão, o que coloca um outro grau de complexidade na definição do plano.

## 6. Conclusão e Trabalhos Futuros

O desenvolvimento distribuído de software apresenta diversas particularidades que ocasionam diversos problemas e proporcionam algumas vantagens na execução das atividades do desenvolvimento de software. Para que seja possível aproveitar as vantagens e mitigar (ou pelo menos gerenciar) os problemas é preciso realizar um planejamento adequado que considere as diferenças dessa forma de desenvolvimento.

Neste artigo foram observados alguns aspectos principais a serem considerados devido ao DDS em um elemento fundamental da garantia da qualidade: o plano de garantia da qualidade. Como base para a discussão utilizou-se um padrão, o IEEE 730, analisando algumas dificuldades principais do DDS a serem tratadas nas diversas seções definidas pelo padrão. Procurou-se com isso demonstrar a complexidade da questão e as preocupações principais que a equipe de garantia da qualidade deve considerar.

No futuro, pretende-se observar empiricamente as dificuldades aqui apontadas durante um estudo de caso que consiste no desenvolvimento de um sistema embarcado feito por uma equipe montada apenas para este projeto e composta por especialistas na área de Astronomia. A preocupação com a garantia da qualidade dentro deste projeto e as dificuldades a serem consideradas para a realização das atividades motivou a discussão realizada neste trabalho. Com isso, pretende-se utilizar as idéias aqui colocadas como base para a criação do plano de garantia da qualidade dentro deste projeto.

Por fim, espera-se que esse trabalho contribua para o entendimento das dificuldades do DDS, algo de suma importância para quaisquer atividades em projetos que empregam essa forma de desenvolvimento.

## Agradecimento

Essa pesquisa foi financiada por uma bolsa de mestrado fornecida pela Escola Politécnica da Universidade de São Paulo e pela CNPq.

## Referências

- ABNT – Associação Brasileira de Normas Técnicas. *NBR ISO/IEC 12207 – Tecnologia de Informação: Processos de Ciclo de Vida de Software*. ABNT, 1998.
- BATTIN, R. D.; CROCKER, R.; KREIDLER, J.; SUBRAMANIAN, K. *Leveraging Resources in Global Software Development*. IEEE Software, v.18, n.2, p.70-77, March/April 2001.
- BIANCHI, A.; CAIVANO, D.; LANUBILE, F.; RAGO, F. VISAGGIO, G. *Towards Distributed GQM*. In: IEEE Workshop On Empirical Studies Of Software Maintenance, 7., Florence, 2001.

- BOURGAULT, M.; LEFEBVRE, E.; LEFEBVRE, L. A.; PELLERIN, R.; ELIA, E. *Discussion of Metrics for Distributed Project Management: Preliminary Findings*. In: Hawaii International Conference On System Sciences, 35., Hawaii, 2002.
- CAIVANO, D.; LANUBILE, F.; VISAGGIO, G. *Scaling up Distributed Software Inspections*. In: Workshop On Software Engineering Over The Internet, 4., Toronto, 2001.
- CARMEL, E. *Global Software Teams*. Prentice Hall, 1999.
- CMMI PRODUCT TEAM. *Capability Maturity Model Integration (CMMI), Version 1.1*. Report CMU/SEI-2002-TR-028. Carnegie Mellon Software Engineering Institute, 2002.
- COCKBURN, A. *Agile Software Development*. Addison Wesley, 2002.
- DAMIAN, D.; ZOWGHI, D. *An insight into the interplay between culture, conflict and distance in globally distributed requirements negotiations*. In: Hawaii International Conference On System Sciences, 36., Hawaii, 2003.
- DUTOIT, A. H. *Communication Metrics for Software Development*. IEEE Transactions on Software Engineering, v.24, n.8, p.615-628, August 1998.
- EBERT, C.; DE NEVE, P. *Surviving Global Software Development*. IEEE Software, v.18, n.2, p.62-69, March/April 2001.
- HAYWOOD, M. *Working in Virtual Teams: A Tale of Two Projects and Many Cities*. IT Professional, v.2, n.2, p.58-60, March/April 2000.
- HEDBERG, H.; HARJUMAA, L. *Virtual Software Inspections for Distributed Software Engineering Projects*. In: International Workshop on Global Software Development, 5., Orlando, 2002.
- HERBSLEB, J.; MOITRA, D. *Global Software Development*. IEEE Software, v.18, n.2, p.16-20, March/April 2001.
- IEEE. *IEEE Guide for Software Quality Assurance Plans*. IEEE Std. 730.1-1995. IEEE, 1995.
- IEEE. *IEEE Standard for Software Quality Assurance Plans*. IEEE Std. 730-2002. IEEE, 2002.
- ISO – International Organization For Standardization. *ISO/IEC TR 15504-1 Software process assessment - Part 1: Concepts and introductory guide*. ISO, 1998.
- KAROLAK, D. W. *Global Software Development: Managing Virtual Teams and Environment*. IEEE Computer Society Press, 1998.
- KOBITZSCH, W.; ROMBACH, D.; FELDMANN, R. L. *Outsourcing in India*. IEEE Software, v.18, n.2, p.78-86, March/April 2001.
- LAYZELL, P.; BRERETON, O. P.; FRENCH, A. *Supporting Collaboration in Distributed Software Engineering Teams*. In: Asia-Pacific Software Engineering Conference, 7., Singapore, 2000.
- MAIDANTCHIK, C.; DA ROCHA, A. R. C. *Managing a Worldwide Software Process*. In: International Workshop On Global Software Development, 5., Orlando, 2002.

- PAULK, M.; CURTIS, B.; CHRISSIS, M. B. WEBER, C. V. *Capability Maturity Model for Software*. Technical Report CMU/SEI-93-TR-024. Carnegie Mellon Software Engineering Institute, 1993.
- PRESSMAN, R. S. *Software Engineering: A Practitioner's Approach*. McGraw Hill, 5a ed., 2001.
- PRIKLADNICKI, R.; YAMAGUTI, M. H. *Risk Management in Global Software Development: A Position Paper*. In: International Workshop On Global Software Development, 3., Edinburgh, 2004.
- SCHULMEYER, G. G.; McMANUS, J. I. *Handbook of Software Quality Assurance*. Prentice Hall, 3a ed., 1998.
- SOMMERVILLE, I. *Software Engineering*. Addison Wesley, 6a ed., 2001.
- UNHELKAR, B. *Process Quality Assurance for UML-Based Projects*. Addison-Wesley, 2003.