

Qualidade em Ecossistemas de Software: Desafios e Oportunidades de Pesquisa

Rodrigo Santos¹, George Valença², Davi Viana³, Bernardo Estácio⁴, Awdren Fontão^{3,5},
Sabrina Marczak⁴, Cláudia Werner¹, Carina Alves², Tayana Conte³, Rafael Prikladnicki⁴

¹PESC/COPPE – Universidade Federal do Rio de Janeiro (UFRJ)

²CIn – Universidade Federal de Pernambuco (UFPE)

³IComp – Universidade Federal do Amazonas (UFAM)

⁴FACIN – Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)

⁵INdT – Instituto Nokia de Tecnologia

{rps, werner}@cos.ufrj.br, {gavs, cfa}@cin.ufpe.br,
{davi.viana, tayana}@icomp.ufam.edu.br, awdren.fontao@indt.org.br
{bernardo.estacio, sabrina.marczak, rafaelp}@pucrs.br

Abstract. *Distributed software development has become a reality over the last decade. The Software Engineering community has investigated technical, economic and social aspects. This scenario leveraged the development of globalized, large-scale and long-term platforms, which is treated as a kernel of software ecosystems. Understanding how this scenario impacts the quality in software ecosystems has been considered a critical success factor. This paper presents an initial discussion on the quality in software ecosystems towards the identification of some challenges and opportunities.*

Resumo. *Desenvolvimento distribuído de software se tornou uma realidade na última década. A comunidade de Engenharia de Software tem investigado aspectos técnicos, econômicos e sociais. Isso propiciou o desenvolvimento de plataformas globalizadas, de larga escala e de longo prazo, que tem sido tratado como o núcleo de ecossistemas de software. Entender como este cenário impacta a qualidade em ecossistemas de software tem sido um fator crítico. O artigo apresenta uma discussão inicial sobre qualidade em ecossistemas de software visando identificar alguns desafios e oportunidades.*

1. Introdução

O Desenvolvimento Distribuído de Software (DDS) se tornou uma realidade na última década (Jiménez *et al.*, 2009). Apesar dos desafios geográficos e culturais da colaboração, uma das principais motivações do DDS está na busca por equipes mais produtivas e especializadas, produtos de melhor qualidade e redução de custos de desenvolvimento (Sengupta *et al.*, 2006). Paralelamente à consolidação do DDS, outros modelos de desenvolvimento têm surgido, não limitados ao escopo de um projeto/produto único (Cataldo & Herbsleb, 2010). Ao reunir diversos projetos em torno de uma tecnologia de software central, esses modelos têm permitido o desenvolvimento de plataformas globalizadas, de larga escala e de longo prazo (Manikas & Hansen, 2013). Tais plataformas originam sistemas mais complexos, que integram uma rede de diversos atores e artefatos, mesmo externos a elas, e formam Ecossistemas de Software (ECOS) (Bosch, 2009). Dittrich (2014) relata que as práticas observadas em ECOS desafiam alguns pressupostos da Engenharia de Software (ES) tradicional, dado que o desenvolvimento dos produtos não ocorre em torno de um projeto e sim pela inovação contínua feita por diversos atores do ecossistema (*i.e.*, inovação aberta). Nesse contexto, um dos fatores críticos do desenvolvimento está na qualidade em ECOS (Santos & Werner, 2012).

Nos diferentes cenários de DDS, fatores que impactam a qualidade são discutidos, seja no desenvolvimento local (*onshoring*) ou global (*offshoring*), dentro da empresa (*insourcing*) ou com terceiros (*outsourcing*) (Prikladnicki & Audy, 2010). Entre eles, destacam-se o alinhamento entre os

processos, as dependências entre as atividades e entre os artefatos, a gestão de projetos globais e a comunicação (Gomes & Marczak, 2012). Estes fatores influenciam na forma como o software é definido, construído, testado e entregue aos consumidores, afetando consequentemente os estágios do ciclo de vida (Jiménez *et al.*, 2009). Semelhante a um dos cenários de DDS, as plataformas de ECOS são desenvolvidas de maneira global, com o envolvimento de terceiros. No entanto, em ECOS, atores externos, conhecidos ou não, atuam no desenvolvimento de produtos e/ou da própria plataforma (Barbosa *et al.*, 2013). Nesse sentido, alguns desafios de DDS podem ser herdados por ECOS, que trazem fatores adicionais para qualidade, como abertura da arquitetura da plataforma, hibridização de modelo de negócio e governança em *outsourcing* (Jansen *et al.*, 2009).

Este artigo apresenta uma discussão inicial sobre qualidade em ECOS visando identificar desafios e oportunidades de pesquisa para a comunidade de ES. Além desta seção de introdução, o artigo está organizado da seguinte forma: a Seção 2 apresenta uma visão geral de como a qualidade tem sido abordada em ECOS; a Seção 3 discute alguns desafios e oportunidades para qualidade em ECOS; e a Seção 4 discorre sobre as considerações finais do artigo.

2. Qualidade em Ecosystemas de Software

Com uma ampla atuação de atores externos à organização que provê a plataforma, como *startups* e desenvolvedores independentes (Miranda *et al.*, 2014), o desafio da qualidade em ECOS vem se tornando uma questão importante (Axelsson *et al.*, 2014). De acordo com Barbosa *et al.* (2013), a questão da qualidade não tem sido definida ou tratada especificamente para ECOS, mas a maneira pela qual a qualidade é medida vem sendo uma preocupação real. As primeiras iniciativas para discutir qualidade em ECOS foram voltadas para a avaliação das contribuições dos atores, o que é crucial para o desenvolvimento e evolução da plataforma (Hmood *et al.*, 2010). Na presença de um *keystone*¹, fatores adicionais são percebidos, como o nível de controle e poder sobre o ECOS (Jansen *et al.*, 2009). Por exemplo, para um ECOS fechado, como o ECOS iPhone, o *keystone* tem controle para garantir que as contribuições atendam a um padrão de qualidade mínimo exigido pelo canal de distribuição (*AppleStore*). Já em um ECOS aberto, como o ECOS MySQL/PHP, onde os atores têm acesso ao código fonte e às bases de conhecimento relacionadas, o controle de qualidade é mais difícil de ser gerenciado e garantido, considerando atores externos desconhecidos.

Além disso, a qualidade em ECOS não tem se restringido à visão dos processos e dos produtos, mas explora a visão de saúde e de prosperidade (Fotrousi *et al.*, 2014). Em (Manikas & Hansen, 2013), percebe-se esta relação com a *saúde* do ECOS, *i.e.*, o grau com que um ECOS oferece oportunidades para colaboradores e para os que dele dependem. Por exemplo, Dhungana *et al.* (2010) discutem dois indicadores: (i) *sustentabilidade*, capacidade do ECOS aumentar ou manter a comunidade de colaboradores diante do aparecimento de tecnologias/produtos de competidores; e (ii) *diversidade*, capacidade do ECOS envolver diferentes colaboradores ao prover uma plataforma que dê suporte a diferentes produtos, linguagens, dispositivos, perfis etc. Já Hartigh *et al.* (2006) apresentam três categorias de indicadores: (i) *produtividade*, capacidade do ECOS adaptar e entregar novas tecnologias, processos e ideias aos seus participantes; (ii) *robustez*, capacidade do ECOS sustentar a sua rede de relacionamentos e manter a arquitetura da plataforma estável; e (iii) *criação de nicho*, capacidade do ECOS inovar ou propiciar que a sua comunidade o faça.

Vale ressaltar que, diferente do processo de desenvolvimento de um único produto, em ECOS não há separação entre o desenvolvimento da plataforma e dos projetos sobre ela (Cataldo & Herbsleb, 2010). Isso requer processos de gestão distribuídos que ampliam a forma de avaliar a

¹Um *keystone* é uma organização que disponibiliza uma tecnologia de software central para ampliar o seu portfólio de produtos e serviços a partir da integração ou construção de novas soluções por atores externos (Jansen *et al.*, 2009).

capacidade e maturidade dos processos dos atores do ECOS (Santos & Werner, 2012). São necessárias estratégias para coordenar diferentes atores e sua comunicação no ECOS (Farias Júnior *et al.*, 2013); técnicas para melhorar a experiência (Fontão *et al.*, 2014) e motivação (Miranda *et al.*, 2014) dos colaboradores; e abordagens (guias e padrões) para certificação das contribuições dos atores (Maia *et al.*, 2013) e para negociar requisitos nos projetos do ECOS (Valença *et al.*, 2013).

3. Desafios e Oportunidades de Pesquisa

Esta seção resume alguns desafios para garantir a qualidade em ECOS. Para identificá-los, utilizou-se a categorização de questões de DDS definida por Audy & Prikladnicki (2008) como orientação: processos de software, infraestrutura, pessoas ou recursos humanos, gerenciamento e comunicação. Os desafios relacionados a DDS (Seção 1) e a explanação sobre qualidade em ECOS (Seção 2) são tomados como base e questões específicas de ECOS são trazidas à discussão:

Categoria 1 – Processos de Software e Padrões

Como responsável pela gestão da plataforma e do ecossistema em torno dela, o *keystone* deve rever seus processos de software para melhorar a experiência dos colaboradores do ECOS (*e.g.*, desenvolvedores externos, fornecedores etc.) e estabelecer controles adequados aos diferentes tipos de atores. Nesse contexto, indicadores de saúde de ECOS podem ser usados como instrumentos para apoiar a avaliação da capacidade e maturidade do *keystone*, visando lidar com a qualidade.

O *keystone* deve disseminar boas práticas e padrões arquiteturais para garantir a consistência das integrações dos produtos no ECOS, cobrindo assim a sustentabilidade, a produtividade e a robustez do ECOS. Seja no contexto de comunidades ou naquele formado por empresas de software, é fundamental que *frameworks* e bibliotecas de desenvolvimento ou sistemas integradores funcionem como mecanismos para garantir o pleno funcionamento das soluções geradas (*e.g.*, capacidade para manter a execução de aplicativos desenvolvidos por terceiros em ECOS móveis).

Categoria 2 – Aspectos Sociais e Gestão do Conhecimento

O conhecimento circulante entre a plataforma e os produtos ao redor influencia o processo de aquisição de tecnologias pelo *keystone*. A diversidade dos elementos que compõem o ECOS e as contribuições intelectuais dos diferentes atores devem ser aproveitadas adequadamente pelos projetos e pela plataforma por meio de diferentes estratégias de captura de conhecimento. Adicionalmente, é importante que a credibilidade de um colaborador seja verificada por meio de certificações ou processos de seleção de parceiros adaptados para as particularidades de ECOS.

Categoria 3 – Governança

Uma estratégia de Gestão dos Produtos de Software (*Software Product Management*) deve ser estabelecida pelo *keystone* e comunicada aos demais atores do ECOS. Desse modo, é possível que haja alinhamento de portfólios e conhecimento do *roadmap* dos produtos. Nesse sentido, sob a ótica de gestão de requisitos, requisitos de qualidade e de integração devem ser identificados e difundidos. Em particular, requisitos de integração podem favorecer a reutilização ao serem incorporados pela plataforma, reduzindo problemas relacionados e facilitando o atendimento de prazos, possibilitando ainda a inovação pela comunidade e ajudando na criação de nicho.

As atividades de governança de ECOS podem auxiliar o *keystone* na identificação de atores e artefatos críticos para a plataforma, internos e externos. Além disso, o *keystone* deve estabelecer orientações quanto a atividades de teste pelos colaboradores de forma a garantir confiabilidade e segurança de seus produtos para diferentes plataformas de ECOS. Por fim, colaboradores precisam analisar e/ou comparar a saúde de diferentes ECOSs de interesse para tomar decisões nos projetos de suas contribuições para a plataforma, *e.g.*, sobre o planejamento de *releases*.

4. Considerações Finais

Este artigo apresentou uma discussão inicial sobre qualidade em ECOS. Alguns dos desafios de qualidade em ECOS identificados são herdados do DDS, porém com uma amplitude de questões relacionadas à arquitetura e modelos de negócio. Entre os itens chave de qualidade, estão a aplicabilidade das metodologias atuais para maturidade e capacidade de processos de software em relação a ambientes de ECOS e a gestão de conhecimento e aspectos sociais dos diferentes atores com a finalidade de promover mais qualidade e saúde para os ECOSs. Além disso, devido à vasta literatura de gestão do conhecimento, devem-se analisar possíveis adaptações de estratégias existentes para os novos ambientes de desenvolvimento gerados pelos ECOSs. Não obstante o amadurecimento das pesquisas mais voltadas para ECOS a partir de 2009 (Barbosa *et al.*, 2013), ainda são necessários trabalhos que investiguem o tópico de forma mais aprofundada. Como próximos passos, pretende-se desenvolver um *framework* para gestão e monitoramento da qualidade em ECOS, bem como explorar a qualidade de processos e produtos a partir de elementos de DDS.

Referências

- Audy, J. & Prikladnicki, R. (2008) “Desenvolvimento Distribuído de Software”. Campus.
- Axelsson, J. *et al.* (2014) “Characteristics of software ecosystems for Federated Embedded Systems: A case study”. Information and Software Technology. In press.
- Barbosa, O. *et al.* (2013) “A Systematic Mapping Study on Software Ecosystems through a Three-dimensional Perspective”. In: Jansen, S. *et al.* (eds.) Software Ecosystems: Analyzing and Managing Business Networks in the Software Industry, Edward Elgar, 59-81.
- Bosch, J. (2009) “From Software Product Lines to Software Ecosystem”. In: SPLC, San Francisco, 1-10.
- Cataldo, M., Herbsleb, J. (2010) “Architecting in Software Ecosystems: Interface Translucence as an Enabler for Scalable Collaboration”. In: 4th ECSA, 2nd IWSECO, Copenhagen, 65-72.
- Dittrich, Y. “Software engineering beyond the project – Sustaining software ecosystems”. Information and Software Technology. In press.
- Farias Junior, I. *et al.* (2013) “Reflexões sobre Comunicação no Desenvolvimento Distribuído no Contexto de Ecosystemas de Software”. In: VII WDDS, Brasília, 52-59.
- Fontão, A. *et al.* (2014) “MSECO Skill: Construção de Competências de Desenvolvedores em Ecosystemas de Software Móvel”. In: XVII CibSE, Pucón, 81-94.
- Fotrousi, F. *et al.* (2014) “KPIs for Software Ecosystems: A Systematic Mapping Study”. In: 5th ICSOB, Paphos.
- Gomes, V., Marczak, S. (2012) “Problems? We All Know We Have Them. Do We Have Solutions Too? A Literature Review on Problems and Their Solutions in GSD”. In: ICGSE, Porto Alegre, 154-158.
- Hartigh, E. *et al.* (2006) “The Health Measurement of a Business Ecosystem”. In: 6th Annual Meeting of the European Chaos and Complexity in Organisations Network, Bergen aan Zee, 39p.
- Hmood, A. *et al.* (2010) “OntEQAM – A Methodology for Assessing Evolvability as a Quality Factor in Software Ecosystems IEEE Industrial Software Evolution and Maintenance Processes”. In: WISEMP, Montreal.
- Jansen, S. *et al.* (2009) “A Sense of Community: A Research Agenda for Software Ecosystems”. In: 31st ICSE, NIER, Vancouver, 187-190.
- Jiménez, M. *et al.* (2009) “Challenges and Improvements in Distributed Software Development: A Systematic Review”. Advances in Software Engineering 2009(3):1-16.
- Maia, N. *et al.* (2013) “Elementos que Impactam o Planejamento de Testes em Ambientes de Desenvolvimento Distribuído no Contexto de Ecosystemas de Software”. In: VII WDDS, Brasília, 60-67.
- Manikas, K., Hansen, K. (2013) “Software Ecosystems – A Systematic Literature Review”. Journal of Systems and Software 86(5):1294-1306.
- Miranda, M. *et al.* (2014) “An exploratory study of the adoption of mobile development platforms by software engineers”. In: 1st MOBILESoft, Hyderabad, 50-53.
- Prikladnicki, R., Audy, J. (2010) “Process models in the practice of distributed software development: A systematic review of the literature”. Information and Software Technology 52(8):779-791.
- Sengupta, B. *et al.* (2006) “A research agenda for DDS”. In: 28th ICSE, Shanghai, 731-740.
- Santos, R., Werner, C. (2012) “ReuseECOS: An Approach to Support Global Software Development through Software Ecosystems”. In: VI WDDS, Porto Alegre, 60-65.
- Valença, G. *et al.* (2013) “Analysing Requirements Negotiation in Software Ecosystems with Multi-Agent Systems Techniques”. In: VII WDDS, Brasília, 44-51.