



Universidade Federal de Pernambuco

Centro de Informática

Ciência da Computação

Análise dos aspectos que afetam o surgimento e desenvolvimento de um Ecossistema de Software

Trabalho de Graduação

Aluna: Amora Cristina Albuquerque Taveira (acat@cin.ufpe.br)

Orientadora: Ph.D. Carina Frota Alves (cfa@cin.ufpe.br)

Co-orientador: Higor Ricardo Monteiro dos Santos (hrms@cin.ufpe.br)

Recife, Junho de 2012

Universidade Federal de Pernambuco

Centro de Informática

Ciência da Computação

Análise dos aspectos que afetam o surgimento e desenvolvimento de um Ecossistema de Software

Trabalho apresentado ao Programa de Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Aluna: Amora Cristina Albuquerque Taveira (acat@cin.ufpe.br)

Orientadora: Ph.D. Carina Frota Alves (cfa@cin.ufpe.br)

Co-orientador: Higor Ricardo Monteiro dos Santos (hrms@cin.ufpe.br)

Recife, Junho de 2012

Assinaturas

Este trabalho de conclusão de curso é resultado dos esforços feitos por Amora Cristina Albuquerque Taveira, orientada pela Ph.D. Carina Frota Alves, intitulado “Analisando os aspectos envolvidos no surgimento e desenvolvimento de um Ecossistema de Software”. As pessoas que assinaram abaixo estão de acordo com o conteúdo deste documento e com os resultados alcançados com este projeto de conclusão de curso.

Carina Frota Alves (Orientadora)

Higor Ricardo Monteiro dos Santos (Co-orientador)

Amora Cristina Albuquerque Taveira (Aluna)

Recife, Junho de 2012

Dedicatória

Dedico este trabalho ao meu Maravilhoso Deus,
e à minha família.

“Mas graças a Deus que nos dá a vitória por
nosso Senhor Jesus Cristo.”

1 Coríntios 15:57

Agradecimentos

Chegar até aqui foi bem mais difícil do que eu pensei e eu não teria conseguido se não tivesse Deus ao meu lado. Em primeiro lugar, eu agradeço a Ele que, durante esses 4 anos e meio de graduação, me fortaleceu e me fez enfrentar com sucesso os maiores obstáculos que passei em toda a minha vida, mesmo quando eu nem se quer percebia a Sua presença.

Esse maravilhoso Deus fez com que eu tivesse sempre ao meu lado as pessoas mais importantes da minha vida, que sempre me deram apoio e fizeram de tudo para que eu permanecesse firme nessa jornada: Paulo Taveira (meu pai); Maria das Graças (minha mãe); Amanda, Marchezan e Suruagy (meus irmãos); Hygia (minha sobrinha); e Kildery (meu noivo). Agradeço também a Rui Neto (meu cunhado), Luana (minha cunhada), e a minha nova família: Lourival (sogro), Mônica (sogra), Kelly (cunhada) e Kennedy (cunhado).

Ele me aproximou também de pessoas que foram verdadeiros amigos na faculdade e que sempre estiveram dispostos a me ajudar, especialmente quando o assunto era programação! Entre elas estão: Adriano, Alice, Anália, Átila, Caio, Denise, Guilherme, Ícaro, Irineu, Ivan, Ivson, Lais, Leonardo (o gordo), Lorena, Ricardo e Ruan. Agradeço também profundamente aos meus amigos de Caruaru que embora eu estivesse muito ausente durante todo esse tempo, não desistiram de mim e estiveram sempre ao meu lado. Em especial: Celinha, Ewila, Felipe Hellen, Merly, Mirtis, Pipo, Susi e Yran. E até a novata Pri. Agradeço aos meus irmãos em Cristo da antiga juventude Batista do Cordeiro que foi a minha primeira família em Recife, e a Igreja Batista Koinonia, a minha mais nova família recifense.

Meus agradecimentos são também para os professores e funcionários do Centro de Informática da UFPE, em especial a minha professora orientadora Carina Frota, ao meu co-orientador Higor Monteiro e Rodolfo Vasconcelos, e aos grupos CITi e PET que me trouxeram um grande crescimento profissional e acadêmico, que vou levar durante toda a minha vida.

Abstract

Currently many organizations decide to create or to be part of a software ecosystem due to the competitive advantage that this approach can offer, due to increased supply base for existing users, the ability to attract new users, the acceleration of innovation and cost sharing. These benefits and opportunities, however, are accompanied by various obstacles that are imposed by this new approach. The focus of this research is to analyze the changes from them to present some aspects which can be influenced or may influence the ecosystem as a whole in a positive or negative way. Knowing these aspects can be the key to minimizing or even to delete the obstacles inherent in a global approach and seize opportunities and benefits brought by it. To facilitate the identification and analysis of these aspects, they will be viewed from three perspectives: social, managerial and business. The social aspects/perspectives will include communication, relationships, motivation and collaboration among members, the managerial aspects involve geographical distance, diverse cultures, different languages and architecture and the business aspects will cover investments, processes and tacit knowledge. To illustrate this study, an example of a real software ecosystem is provided at the end of this document.

Resumo

Atualmente diversas organizações decidem criar ou fazer parte de um ecossistema de software devido ao diferencial competitivo que esta abordagem pode oferecer, resultante do aumento da oferta básica para os usuários existentes, da capacidade de atrair novos usuários, da aceleração da inovação e do compartilhamento de custos. Tais benefícios e oportunidades, no entanto, são acompanhados por diversos obstáculos que são impostos por esta nova abordagem. O foco desta pesquisa é analisar as mudanças ocorridas para a partir delas apresentar alguns aspectos que podem ser influenciados, ou podem influenciar o ecossistema como um todo de maneira positiva ou negativa. Conhecer tais aspectos pode ser a chave para minimizar ou até mesmo excluir os obstáculos inerentes a uma abordagem global, e aproveitar as oportunidades e benefícios trazidos por ela. Para facilitar a identificação e análise desses aspectos, eles serão vistos sob três perspectivas: social, gerencial e de negócio. Os aspectos sociais irão englobar a comunicação, os relacionamentos, a motivação e a colaboração entre os membros; os aspectos gerenciais envolverão distância geográfica, culturas diversificadas, diferentes linguagens e arquitetura; já os aspectos de negócio irão abranger os investimentos, processos e o conhecimento tácito. Para ilustrar este estudo, um exemplo de ecossistema de software real é apresentado ao final deste documento.

Palavras-chave: Ecossistemas de Software, Surgimento de um SECO, Aspectos Sociais, Aspectos Gerenciais, Aspectos de Negócios.

Sumário

Lista de Figuras.....	ii
Lista de Quadros	iii
1. Introdução.....	1
1.1. Contexto e Motivação	1
1.2. Objetivos	2
1.3. Estrutura do Documento	3
2. Ecossistemas de Software	5
2.1. Definições.....	5
2.2. Características dos Ecossistemas de Software	8
2.2.1. Papéis Envolvidos no Ecossistema	12
2.2.2. Ciclo de Vida de um SECO	15
3. Surgimento de um SECO	17
3.1. Aspectos que afetam o surgimento e desenvolvimento de um SECO	21
3.1.1. Aspectos Sociais	23
3.1.2. Aspectos Gerenciais.....	28
3.1.3. Aspectos de Negócios	32
4. Ecossistema do iPhone.....	37
5. Conclusões e Trabalhos Futuros.....	44
Referências	48

Lista de Figuras

Figura 1. Complexidade das relações e nível de turbulência e inovação relacionados com os papéis dos ecossistemas.	15
Figura 2. Ciclo de vida de um Ecossistema de Software	16
Figura 3. Aspectos que envolvem o surgimento e o desenvolvimento de um SECO. ..	36
Figura 4. Demanda de desenvolvimento de aplicativos para iPhone.	40
Figura 5. IDE para desenvolvimento de aplicativos.	40
Figura 6. Visão geral do ecossistema do iPhone.	42

Lista de Quadros

Quadro 1. Comparação entre os tipos de ecossistemas.	12
Quadro 2. Os primeiros parceiros do ecossistema do iPhone.	38

1. Introdução

Este capítulo tem o propósito de apresentar as definições de Ecossistemas de Software e o contexto no qual este conceito está inserido, além de mostrar os objetivos requeridos por este trabalho e a forma como este documento está estruturado.

1.1. Contexto e Motivação

Para diversas organizações, o desenvolvimento de software em larga escala é complicado, caro, lento e imprevisível. Quatro décadas de pesquisa em engenharia de software tem resultado numa ampla gama de técnicas para gerenciar a complexidade do desenvolvimento de sistemas de software, porém o crescimento cada vez maior desses sistemas mais modernos acarreta a necessidade de novas abordagens para gerenciar esta complexidade (BOSCH e BOSCH-SIJSEMA, 2010). Além disso, os projetos de software, em sua maioria, são dependentes de outros projetos, sendo então raramente desenvolvidos de forma autônoma.

Empresas de software já não funcionam como unidades independentes que podem fornecer produtos separados. Estas se tornaram dependentes de outras empresas no que se refere ao compartilhamento de componentes de software e infraestrutura, tais como sistemas operacionais, bibliotecas, lojas de componentes e plataformas. Devido à rápida mudança de tecnologia e à alta complexidade no desenvolvimento de software em larga escala, estes fornecedores recorrem à integração virtual através de alianças para estabelecer redes de influência e interoperabilidade. Estas redes são chamadas Ecossistemas de Software (SECO's), um conceito que se tornou vital para explicar a vida e a morte de fornecedores de software e suas tecnologias (METZGER, 2001).

O termo Ecossistema de Software pode ser definido como uma rede informal de unidades, que influenciam umas as outras de forma positiva quando se trata de sucesso econômico e outros benefícios. Assim como nos Ecossistemas Biológicos, cada empresa ou unidade em um Ecossistema de Software compartilha o seu destino com as

demais empresas (*players*) existentes no mesmo ecossistema. A saúde de um ecossistema é, portanto, determinada pela saúde dos *players* existentes nele.

Devido às complexas e diversas relações que podem existir entre os parceiros, e às mudanças causadas pela transição de uma abordagem organizacional mais centralizada para uma abordagem descentralizada na qual diferenças funcionais, geográficas e culturais surgem, muitas barreiras poderão aparecer durante o desenvolvimento de um ecossistema de software. Caso essas dificuldades não sejam trabalhadas de forma a excluí-las ou, pelo menos, atenuá-las, a saúde do ecossistema poderá ser diretamente afetada de forma negativa, prejudicando, portanto, a rede existente de parceiros. Por outro lado, se tais diferenças forem trabalhadas de maneira satisfatória juntamente com as potenciais oportunidades, fazer parte de um SECO pode resultar em grandes benefícios e diferenciais competitivos que devem ser aproveitados.

1.2. Objetivos

No contexto apresentado na seção anterior, o objetivo principal deste trabalho é investigar quais as mudanças ocasionadas e os aspectos que devem ser considerados quando uma organização decide criar ou fazer parte de um ecossistema de software. Tendo esta informação, é possível saber como se deve lidar com essas mudanças de forma que elas não prejudiquem o desenvolvimento do ecossistema, e fazer com que as oportunidades advindas desta abordagem possam ser aproveitadas.

De maneira geral, esta pesquisa apresentará as definições para Ecossistemas de Software (SECO), acrescidas dos conceitos de Ecossistema de Negócios e da própria definição de Ecossistemas Biológicos. Além de mostrar a forma como se dá o surgimento de um SECO e as mudanças pelas quais eles ficam suscetíveis. Ela irá também especificar e analisar os aspectos que influenciam ou são influenciados, direta ou indiretamente, pelos Ecossistemas de Software a partir do seu surgimento, e agrupar estes aspectos de acordo com a esfera sobre a qual ele terá maior impacto: Social, Gerencial ou Negócio. Tais aspectos devem ser trabalhados de modo a mitigar, ou até mesmo excluir os obstáculos que surgem com a mudança de uma abordagem tradicional para uma abordagem de ecossistemas adotada por um conjunto de organizações,

podendo também aumentar o aproveitamento das oportunidades existentes. Informações desse tipo podem ser úteis aos parceiros, em especial ao *keystone*, já que se não consideradas podem causar a não rentabilidade do ecossistema, e conseqüentemente a sua morte. Por outro lado, se tais aspectos forem considerados, os parceiros poderão usufruir dos diversos benefícios gerados pela adoção dessa abordagem.

Como exemplo de estudo, este trabalho irá abordar o ecossistema do iPhone, descrevendo o seu surgimento e como Apple, organização *keystone*, e o ecossistema como todo lidam com os aspectos inerentes à abordagem de ecossistemas, de forma a obter sucesso e alta lucratividade.

1.3. Estrutura do Documento

Este documento é estruturado como segue:

- Capítulo 2 (Ecossistemas de Software): Este capítulo apresenta as definições gerais de ecossistemas de software, juntamente com suas características, ciclo de vida e papéis representados pelas organizações envolvidas;
- Capítulo 3 (Surgimento de um SECO): Neste capítulo a forma como se dá o surgimento de um ecossistema de software é explicada, bem como os aspectos que devem ser considerados neste tipo de abordagem, agrupados em aspectos sociais, gerenciais e de negócio;
- Capítulo 4 (Ecossistema do iPhone): Neste capítulo o SECO do iPhone é apresentado. A maneira com surgiu, os parceiros envolvidos, como se deu a disponibilização da plataforma para desenvolvedores externos, e como a Apple coordena todo esse ecossistema lucrativo são tópicos abordados neste capítulo;
- Capítulo 5 (Conclusões e Trabalhos Futuros): Este capítulo apresenta as considerações sobre o conteúdo mostrado neste documento, bem como as

questões que ainda estão em aberto neste estudo, e que podem ser abordadas em trabalhos futuros.

2. Ecossistemas de Software

Este capítulo tem como objetivo apresentar as definições e característica dos ecossistemas de software, incluindo as que foram herdadas dos ecossistemas biológicos e de negócios. Serão apresentados, também, o ciclo de vida de um SECO e os papéis que as organizações poderão representar dentro desta comunidade.

2.1. Definições

O conceito de ecossistemas de software foi herdado do conceito de ecossistemas ecológicos, que pode ser definido como o conjunto formado por todas as comunidades que vivem e interagem em determinada região, e pelos fatores abióticos que atuam sobre essas comunidades (água, sol e solo, por exemplo). Em outras palavras, é um conjunto de comunidades interagindo entre si e agindo sobre os fatores abióticos, ou sofrendo a ação deles (WRI, 2000). De acordo com Yu et al. (2008), da perspectiva evolutiva, o ecossistema é uma organização dinâmica que compreende todas as respostas dos habitantes entre si e seu meio ambiente compartilhado.

As principais características dos SECO's estão fundamentadas nestes conceitos da biologia referentes a ecossistemas biológicos. A partir deles, Moore (1993) introduziu o conceito de Ecossistema de Negócios, que também serviu como base para a fundamentação de SECOs. Ele afirma que ecossistema de negócios é um conjunto de organizações a qual fazem parte diversas indústrias, e que trabalham cooperativamente e competitivamente em termos de produção, serviço ao cliente e inovação. Neste ecossistema, companhias co-evoluem em suas capacidades em volta de alguma inovação. Elas trabalham de forma mútua para suportar novos produtos, satisfazer as necessidades dos clientes e, eventualmente, incorporar outras inovações.

Moore (1993) ainda ressalta que Ecossistemas de Negócios, assim como na biologia, gradualmente se movem de uma coleção aleatória de elementos para uma comunidade mais estruturada. Eles condensam o capital, os interesses do cliente, e o talento gerado por uma inovação, com o mesmo sucesso que as espécies o fazem a partir

dos recursos naturais da luz solar, água e nutrientes do solo. Como nos Ecossistemas Biológicos, eles se desenvolvem através de auto-organização, emergência e co-evolução que ajudam a adquirir a capacidade de adaptação. Além disso, estão presentes também a competição e a cooperação agindo simultaneamente. Da mesma forma que um indivíduo de uma determinada espécie em um ecossistema biológico, cada membro de um ecossistema de software ou de negócios, em última análise, compartilha o destino de sua rede como um todo, independentemente da sua força aparente. Tais conceitos podem ser levados também para os ecossistemas de software. Outra característica herdada é o fato desses ecossistemas estarem localizados num ambiente onde diversos aspectos podem trazer impacto na adaptação dos mesmos, e estes também podem impactar esses aspectos. São exemplos deles: política, sociedade, cultura e aspectos jurídicos.

Ecossistemas de software estão cada vez mais ganhando popularidade entre as grandes organizações devido aos diversos benefícios trazidos por este conceito, tais como compartilhamento de custos e aumento de produtividade, o que leva a um significativo diferencial em termos de competitividade. É notável que sistemas de software não funcionam de forma independente, ou seja, as aplicações de software normalmente trabalham em conjunto com sistemas operacionais, sistemas de banco de dados e/ou sistemas de redes. Além disso, esses sistemas de software não só dependem de elementos computacionais, mas também de fatores externos ao ambiente de software, tais como clientes, usuários, e mercados. Yu et al. (2008) citam que ecossistema de software se refere a uma coleção de produtos de software que apresentam um determinado grau de relações simbióticas. Relações simbióticas, de acordo com a biologia, implicam num tipo de inter-relação de tal forma íntima entre os organismos envolvidos que se torna obrigatória. É uma relação vantajosa para pelo menos um dos organismos vivos de espécies diferentes na qual eles agem ativamente, não podendo ficar separados um do outro ou, do contrário, pelo menos um deles morreria (GRABIANOWSKI, 2008).

Campbell e Ahmed (2010) afirmam que, em seu sentido mais fundamental, um ecossistema de software é um sistema no qual as paredes tradicionais entre as entidades de desenvolvimento são quebradas, permitindo a colaboração e interoperabilidade entre as partes. SECO's referem-se à forma como as empresas interagem para criar e entregar

os produtos, de forma que cada parceiro, dentro de uma rede interligada de empresas, tem um papel e desempenho individual que influenciam a saúde do ecossistema como um todo (ANDRADE et al., 2011).

Pesquisas sobre SECO's vêm se mostrando bastante úteis dentro do contexto de software, visto que projetos de software são considerados sofisticados e complexos. Para que estes projetos sejam lançados no mercado, existem diversas atividades envolvidas tanto a nível técnico, como a nível de negócio. Como parte do ciclo de vida do desenvolvimento de software, todo software necessita de personalizações, integrações e/ou suporte. Andrade et al. (2011) definem ecossistemas de software como uma plataforma de software, um conjunto de desenvolvedores internos e externos, e uma comunidade especialista no domínio em serviços para o conjunto de usuários, que compõe soluções relevantes para as suas necessidades. Dessa forma, o campo de abordagem é bem mais amplo que no escopo de linha de produção intra-organizacional, visto que desenvolvedores externos, extensões de fornecedores e outras contribuições externas fazem parte do desenvolvimento do software. Por isso, ecossistemas de software criam dependências que antes não existiam entre os componentes e as organizações associadas.

Há, ainda, outras definições para SECO's, entre elas, Boucharas et al. (2009) os cita como uma rede informal de unidades (legalmente independentes), que influenciam umas as outras de forma positiva quando se trata de sucesso econômico e outros benefícios; ou ainda como um conjunto de atores funcionando juntos como uma unidade e interagindo com um mercado compartilhado para softwares e serviços. Essas relações são frequentemente sustentadas por um mercado ou por uma única plataforma tecnológica e operam através da troca de informações, recursos e artefatos.

Apesar de ecossistemas de softwares só agora estiverem recebendo uma maior atenção em termos de estudo e pesquisa, esta categoria de ecossistemas existe há algumas décadas. Sua importância foi confirmada especialmente no início de 1990 quando diversas empresas brigavam pelo domínio de sistemas operacionais de desktop, e ganhavam o apoio cada vez maior em termos de quantidade e influência de desenvolvedores externos. Este apoio foi reconhecido como crucial e empresas como a IBM e Microsoft conseguiram explicitamente atingir esta dimensão (BOSCH, 2009).

Os SECOs na maioria das vezes se baseiam em uma plataforma comum sobre a qual diferentes parceiros contribuem com suas próprias inovações. Além disso, o custo de desenvolvimento é compartilhado entre as empresas envolvidas com a plataforma, que ficam livres para estendê-la com seus próprios módulos (KILAMO et al., 2011). Esta definição está estreitamente relacionada ao conceito de Open Source Software (OSS), conhecido também como Software Livre. Este tipo de software normalmente é iniciado a partir da decisão de uma companhia de compartilhar a sua plataforma de desenvolvimento com desenvolvedores externos à mesma. Weiss (2011) ressalta que empresas são motivadas a tomarem esse tipo de decisão pelo diferencial que elas podem alcançar quando comparada a seus concorrentes, já que estatísticas mostram que entre 50% e 90% do esforço de desenvolvimento de software é gasto com a criação de software que não traz qualquer diferencial a empresa que o desenvolveu. Transparência no desenvolvimento e liberdade para cada vez mais construir sistemas mais complexos a partir dos blocos de desenvolvimento já disponíveis são aspectos que normalmente são comuns a softwares open source, que acabam por fornecer uma forma única de construir ecossistemas rapidamente, sem um grande investimento inicial (KILAMO et al., 2011).

2.2. Características dos Ecossistemas de Software

Como visto na seção 2.1, existe uma analogia considerável entre os ecossistemas biológicos e os ecossistemas de negócios e de software. Todos eles são caracterizados por um grande número de participantes, interligados entre si, que dependem um do outro para sua eficácia e sobrevivência. Cada membro de um ecossistema compartilha o destino da rede como um todo, independente de sua força aparente. Isso se dá porque cada vez mais as empresas, produtos e tecnologias de uma rede estão cada vez mais interligados em relacionamentos mutuamente dependentes, como as espécies em um ecossistema biológico (MOORE, 1993). Na biologia, os indivíduos podem interagir de forma harmônica ou desarmônica. Relações harmônicas definem as interações entre duas espécies nas quais ambas se beneficiam ou pelo menos uma se beneficia e a outra não é afetada. Enquanto que relações desarmônicas são caracterizadas por pelo menos uma das espécies envolvidas sofrer algum prejuízo, mesmo que a outra seja beneficiada.

Os tipos mais comuns de relações entre indivíduos na biologia e que podem ser trazidas para os ecossistemas de software são:

- **Mutualismo:** Relação harmônica, na qual ambos os indivíduos se beneficiam. É de caráter obrigatório, ou seja, os indivíduos não conseguem sobreviver sem ela;
- **Comensalismo:** Relação harmônica, na qual apenas um indivíduo é beneficiado, enquanto que o outro não é afetado;
- **Parasitismo:** Relação desarmônica na qual um indivíduo se beneficia com a relação, e o outro é prejudicado. Tal relação pode causar a morte do indivíduo que foi afetado;
- **Amensalismo:** Relação desarmônica na qual um indivíduo é prejudicado, embora o outro não seja afetado;
- **Competição:** Ambos os indivíduos envolvidos são prejudicados;
- **Neutralismo:** Nenhum dos indivíduos é afetado com o relacionamento.

Segundo Yu et al. (2008), nem todas essas relações são aparentes em SECOs. As mais comuns são mutualismo, comensalismo e competição. Outro tipo de relação existente na biologia e que foi trazida para os SECOs é a simbiose. O conceito tradicional de simbiose equivale ao de mutualismo, no entanto, alguns biólogos consideram que qualquer relação entre espécies envolvendo freqüente contato íntimo é simbiose, incluindo comensalismo e parasitismo - relações nas quais um dos indivíduos não é afetado, ou um deles é prejudicado, respectivamente (GRABIANOWSKI, 2008). Yu et al. (2008) fazem a analogia com ecossistemas de software a partir desse conceito geral de simbiose e afirmam que biólogos consideram que esse tipo de relação é a força motriz para a evolução de ecossistemas. Eles afirmam também que a cooperação entre indivíduos é um fator essencial no conceito de seleção natural, sendo selecionados, frente às mudanças ambientais, àqueles indivíduos que aprenderam a cooperar com os demais dentro do ecossistema. Essa importância da cooperação e mutualismo é trazida também para os SECOS, considerando que empresas devem trabalhar junto com fornecedores e parceiros, contribuindo uma com as outras, para conseguirem atingir a

máxima cobertura de mercado. Andrade et al. (2011) enfatizam não só a cooperação, mas também a competição, afirmando que ambas suportam a criação de novos produtos, a satisfação das necessidades dos clientes e a inovação.

Outras características inerentes aos ecossistemas são as de co-evolução, auto-organização e emergência. A co-evolução está relacionada com o potencial efeito de sucesso que os parceiros têm uns sobre os outros. Sendo assim, para que um *player* evolua, é necessária a evolução simultânea de outros *players*. A auto-organização é caracterizada pelo controle descentralizado dos ecossistemas. Tendo esta característica, as organizações envolvidas no SECO, embora tenham um conhecimento limitado para a tomada de decisões, possuem autonomia para atuar sem a existência de um controle externo. Essa descentralização será melhor explicada na seção 3.1.2. Por fim, o conceito de emergência se refere à característica de que medidas e ações executadas dentro do SECO levam a conseqüências imprevisíveis no comportamento dos membros existentes nele. Ainda que a emergência esteja associada a indivíduos, ela influencia o ecossistema como um todo.

Há outro conceito de significativa importância herdado dos ecossistemas biológicos para os ecossistemas de negócios e de software, que se refere à sustentabilidade e prosperidade dos ecossistemas: é o conceito de saúde. De acordo com Iansiti e Levien (2004), a saúde de um ecossistema é forma que se tem de avaliar a sua força em um momento específico. Algumas perspectivas são definidas a fim de medir a saúde de um ecossistema de negócios, podendo também ser utilizadas na abordagem de SECOs. São elas:

- **Robustez:** Assim como na natureza, ecossistemas de negócio e de software sofrem choques internos e externos. Tais ecossistemas devem estar protegidos contra mudanças imprevistas que possam vir a ocorrer no ambiente na qual eles estão inseridos. Uma empresa que faz parte de um ecossistema robusto goza de relativa previsibilidade, e as relações entre os membros do ecossistema estão melhor protegidas contra esses possíveis choques. Um exemplo de medida de robustez é a taxa de sobrevivência dos membros, que pode ser medida ao longo do tempo, ou em relação a outros ecossistemas.

- **Produtividade:** Iansiti e Levien (2004) citam que é a medida mais importante na saúde dos ecossistemas biológicos é a sua habilidade de converter fatores abióticos, tais como luz do sol e nutrientes minerais em fatores bióticos, como população de organismos. Eles dizem que nos negócios tal medida é equivalente a habilidade da rede em transformar tecnologia e outras matérias primas de inovação em baixos custos e novos produtos. O retorno do capital investido é um dos caminhos para se medir a produtividade.

- **Criação de Nicho:** descreve a capacidade do ecossistema de criar novas oportunidades para os novos e velhos atores, fornecendo novas oportunidades de negócios (JANSEN et al., 2009). Iansiti e Levien (2004) dizem que a literatura ecológica indica a importância de os ecossistemas possuírem certa diversidade de espécies. A melhor medida para criação de nicho está na capacidade que o ecossistema tem de aumentar esta diversidade através da criação de novas funções. Para estimar a criação de nicho deve-se analisar se as tecnologias emergentes estão realmente sendo aplicadas sob a forma de variedade de novos produtos comerciais.

Tomando como base todos esses conceitos herdados da biologia, Barbosa (2012) apresenta um paralelo entre os contextos de ecossistemas biológicos e de software. Tais relações são apresentadas no Quadro 1.

Característica	Ecossistemas Biológicos	Ecossistemas de Software
Reservatório ou recursos finitos	Recursos críticos como energia (alimento), minerais (nutrientes), ou água são gerenciados implicitamente pela lei da natureza.	Os recursos críticos são a arquitetura da plataforma, código, tempo, dinheiro, usuários, desenvolvedores, etc. Frameworks regulatórios são necessários para gerenciar esses recursos apropriadamente.
População	É controlada pelos fatores que influenciam positiva ou negativamente a incorporação ou permanência dos indivíduos no ecossistema (por exemplo, melhores condições para reprodução e disponibilidade de alimentos).	Uma das influências são os fóruns de incentivo para usuários que compram, usam ou estendem os produtos do ecossistema de software.
Transferência de energia	Transferência de energia e material de um grupo para outro.	Transferência e interpretação de conhecimento entre diferentes stakeholders no ecossistema de software.
Adaptação dos indivíduos e de todo o sistema às mudanças	Ecossistemas naturais são caracterizados como ambientes onde indivíduos podem adaptar seu comportamento de acordo com as mudanças do sistema em que eles vivem.	Existe a responsabilidade dos stakeholders de aceitar, rejeitar ou “aprender a conviver” com os produtos de software do ecossistema.
Processos	Definidos a partir de ciclos de vida encontrados nos ecossistemas naturais (tamanho da população e biodiversidade)	Definidos a partir de ciclos de mercado e avanços tecnológicos.

Quadro 1. Comparação entre os tipos de ecossistemas.

2.2.1. Papéis Envolvidos no Ecossistema

Como visto, ecossistemas são formados por um conjunto de atores funcionando juntos como uma unidade e interagindo com um mercado compartilhado para softwares e serviços. No contexto de ecossistemas, cada ator, ou *player*, possui um papel

específico a ele associado. Desses papéis definidos no ecossistema, segundo Jansen et al. (2009), ao longo do tempo dois deles podem ser reproduzidos com segurança por organizações: *keystone* e *niche player*.

Na biologia, o *keystone* é definido como uma espécie que tem um efeito desproporcional sobre o seu ambiente em relação a sua quantidade existente no mesmo. Isso significa que caso esse tipo de espécie seja removida do ecossistema, embora ela esteja presente em uma pequena quantidade em termos de biomassa ou produtividade, esse ecossistema sofrerá uma mudança dramática (POWER et al., 1996). Estas espécies têm, portanto, um papel crítico na manutenção da estrutura de uma comunidade ecológica, afetando muitos outros organismos e ajudando a determinar os tipos e números de várias outras espécies dentro da comunidade.

Jansen et al. (2009) dão o exemplo do jaguar como um *keystone* que, através da grande variedade de espécies que persegue, ajuda a equilibrar o ecossistema florestal dos mamíferos. Nos SECOs, o papel do *keystone* é bastante semelhante ao da biologia, e se trata da organização que fornece um padrão ou plataforma tecnológica, provendo a base para o desenvolvimento do ecossistema. Esses autores consideram que os papéis do ecossistema são formados por *shapers* e *participants*. O conceito de *shapers* é equivalente ao do *keystone* e para esse tipo de papel existem três necessidades:

- **Visão do *keystone*:** É identificada pelo fornecimento de foco, de oportunidades e descreve as forças fundamentais para os outros *players* envolvidos no SECO;
- **Plataforma:** A plataforma fornecida pela organização *keystone* deve promover o desenvolvimento do ecossistema, definir padrões e práticas a serem seguidas, fornecer especialização, e aumentar o seu valor com o aumento do número de participantes;
- **Comportamento de *keystone*:** O *keystone* deve ter considerável credibilidade dentro do SECO, limitar os riscos para as novas aquisições e mostrar aos outros membros da comunidade que eles poderão esperar, de sua parte, um compromisso em longo prazo.

De acordo com Jansen et al. (2009), esses três pilares têm como objetivo melhorar a saúde geral do ecossistema, fornecendo um conjunto estável de bens comuns que as

demais organizações utilizarão para construir suas próprias contribuições. Em síntese, *keystones* criam e compartilham valor dentro do SECO.

Niche players fazem parte da maioria dos *players* existentes no ecossistema. Eles ficam em busca de reforçar a sua capacidade em certas atividades e fazem uso dos recursos fornecidos pelo *keystone*, utilizando a sua tecnologia padrão ou plataforma, contribuindo para geração de valor dentro do ecossistema através de suas funções especializadas, sendo, portanto, mutuamente dependentes deste, ou ainda de outros *niche players*. De acordo com Iansiti e Levien (2004), tais organizações objetivam desenvolver capacidades especializadas que a diferenciam das demais companhias dentro da rede. Ao contrário do *keystone*, *niche players* representam a massa dos ecossistemas e são responsáveis pela maior parte da geração de valor e de inovação dentro da comunidade.

O terceiro papel apresentado por Jansen et al. (2009) é o de *dominator*, organismos que progressivamente absorvem ou eliminam outros organismos (*keystones*, *niche players* ou outros *dominators*). As organizações que desempenham este papel dentro do ecossistema tentam controlá-lo, fazendo uso de recursos de outras companhias sem um benefício recíproco, sendo, portanto, consideradas como exploradoras. Tais companhias podem até ser bem sucedidas inicialmente, mas à medida que elas eliminam as outras organizações existentes, diminuindo a massa crítica da comunidade, tendem a se tornar as únicas fornecedoras de inovação dentro do ecossistema, ao mesmo tempo em que devem fornecer também a capacidade suficiente para atender toda a base de clientes existente. A existência de um papel deste tipo no ecossistema faz com que muitos parceiros não sejam atraídos a se juntar ao mesmo.

A Figura 1 foi retirada de Iansiti e Levien (2004) e mostra os níveis de turbulência, inovação e complexidade das relações associados aos papéis existentes dentro dos ecossistemas.

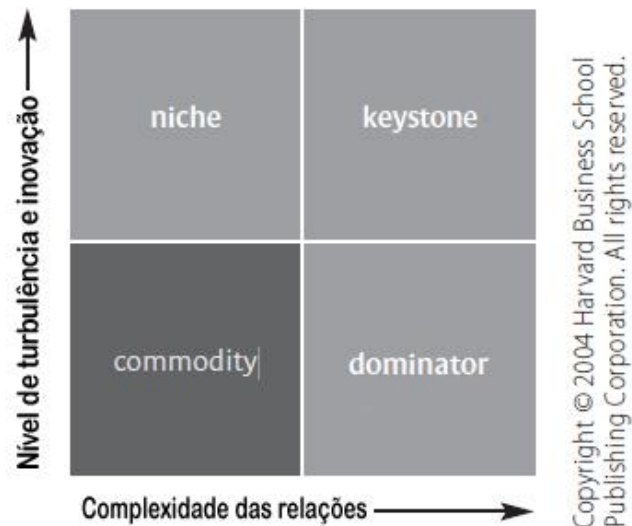


Figura 1. Complexidade das relações e nível de turbulência e inovação relacionados com os papéis dos ecossistemas.

2.2.2. Ciclo de Vida de um SECO

Inspirado no modelo evolucionário de ecossistemas de negócios propostos por Moore (1993), Andrade et al. (2011) descreve o ciclo de vida de um SECO, apresentado na Figura 2, contendo quatro estágios:

- **Nascimento:** Para que seja dado início às parcerias entre as organizações, é necessário que haja uma proposta inicial fornecida pela organização *keystone* para atender a alguma demanda de mercado. Caso a proposta fornecida por esta organização seja promissora, *niche players* serão atraídas por ela, formando então as parcerias necessárias ao desenvolvimento da rede. Alguns fatores tais como capital, potencial de inovação, demanda de clientes e reputação agem como alguns dos atrativos para que empresas se juntem ao *keystone* e dêem início ao ecossistema.

- **Expansão:** Envolve as lutas para se conquistar novos clientes e estimular a demanda de mercado.

- **Liderança:** Nesta etapa, disputas ocorrem dentro do ecossistema entre organizações que desejam alcançar a sua liderança. Tais disputas ocorrem

especialmente se o SECO for lucrativo. Embora a concorrência entre os participantes do ecossistema possa ser intensa, a cooperação deve continuar a existir tendo em vista a sua importância para a sustentabilidade do mesmo. Nesse estágio, o próprio *keystone* pode perder sua liderança. Os autores afirmam que para se defender da concorrência e possível perda de poder, os participantes precisam fortalecer seu poder de negociação e suas alianças. Tal estratégia implica em contribuir ativamente para a saúde do SECO, e fornecer soluções inovadoras diferenciadas das que os outros *players* podem oferecer.

- **Auto-renovação:** Nesta etapa, os parceiros devem trazer para o ecossistema inovações que possam mantê-lo forte e adaptável a mudanças, ou seja, robusto. Tais mudanças podem ser em termos de regulamentos, economia, ou até mesmo do surgimento de um concorrente potencial. O SECO corre o risco de morrer caso essas mudanças não sejam levadas em consideração.

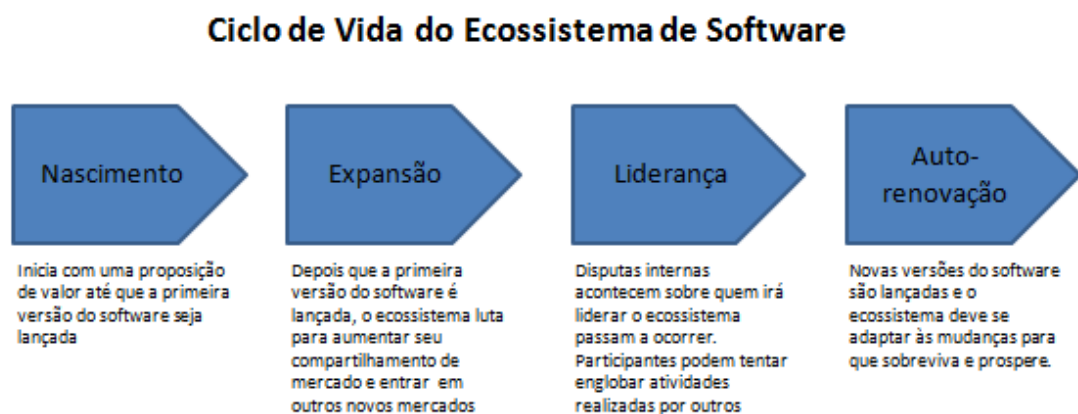


Figura 2. Ciclo de vida de um Ecossistema de Software

Moore (1993) prevê que à medida que a abordagem ecológica para gestão se torna mais comum, cada vez mais os executivos se tornam conscientes da co-evolução e de suas conseqüências, fazendo com que o ritmo de mudança dos negócios acelere. Já os executivos que estão limitados pelas perspectivas da indústria tradicional não enxergarão as oportunidades futuras e os reais desafios aos quais as suas organizações estão submetidas.

3. Surgimento de um SECO

De acordo com Bosch (2009), uma organização não deve se limitar ao sucesso de uma linha de produtos de software, a empresa deve continuar em expansão além dos seus limites intra-organizacionais, disponibilizando sua plataforma a partes externas à empresa. A partir do momento que a empresa torna disponível a sua plataforma além dos limites organizacionais, ocorre a transição de uma linha de produção de software para um ecossistema de software.

Existem diversos benefícios que uma companhia pode adquirir quando ela torna sua plataforma disponível externamente. Bosch (2009) cita duas das principais motivações. A primeira delas é que a empresa pode perceber que a quantidade de funcionalidades que precisam ser desenvolvidas para satisfazer as necessidades de seus clientes é muito mais do que aquilo que pode ser desenvolvido com uma quantidade razoável de tempo e com um significativo investimento em P&D que ofereça um retorno aceitável sobre o investimento. Ele ressalta que o mercado muitas vezes opera com base no “*winner takes all*”, o vencedor leva tudo. Sendo assim, a construção de uma grande base de clientes o mais rapidamente possível é uma estratégia fundamental para o sucesso a longo prazo. A segunda motivação é que a tendência de personalização em massa impulsiona a necessidade de um investimento significativo em P&D, e os usuários exigem cada vez mais um grau significativo de personalização, especialmente para serviços web e dispositivos móveis. Estender o produto para desenvolvedores externos facilita a customização em massa. As empresas que inicialmente construíram alguma aplicação web de sucesso, por exemplo, são levadas a criar a plataforma dessa aplicação e disponibilizá-la para desenvolvedores externos, já que eles podem fornecer funcionalidades que atendem às necessidades dos segmentos de usuários, as quais a empresa que disponibiliza a plataforma seria incapaz de fornecer por si só.

Em síntese, as empresas optam por disponibilizar sua plataforma para terceiros, entre outros motivos, devido ao aumento da oferta básica para os usuários existentes, à capacidade de atrair novos usuários, à aceleração da inovação e ao compartilhamento de

custos dessa inovação. Seguindo essa tendência, de acordo com Kilamo et al. (2011), cada vez mais as empresas estão lançando o seu software proprietário como código aberto, formando um ecossistema de software de projetos de desenvolvimento relacionados, complementado com um ecossistema social dos membros das comunidades. Do ponto de vista de engenharia, um ecossistema de software fornece a tecnologia para implementação, o ambiente para a infraestrutura global de software e uma metodologia de desenvolvimento. Esses autores destacam a evidência de que aproveitar um sistema já existente como base para um ecossistema promete grandes oportunidades de negócios que devem ser aproveitadas.

O primeiro passo importante para se dar início a um SECO, é uma organização adotar uma linha de produtos de software. O conceito de linhas de produto de software consiste em um conjunto de características em comum que satisfazem uma necessidade específica de um domínio, e que são desenvolvidas a partir de um núcleo comum conforme prescrito (CLEMENTS, 2005). Bosch (2009) afirma que as linhas de produtos de software certamente podem ser vistas como a abordagem intra-organizacional mais bem sucedida no que se refere à reutilização de software. Entre outros benefícios, tal abordagem tornou possível a melhora significativa da eficiência de P&D nas empresas e o aumento do portfólio de produtos existente. Devido a essas vantagens, em termos de negócio, linhas de produto de software têm um impacto positivo importante nas empresas que conseguem aplicar essa tecnologia com êxito.

De acordo com Bosch e Bosch-Sijtsema (2010), há um caminho claro no que se refere ao surgimento de um SECO a partir de linhas de produto de software, já que nesta última existe uma plataforma a partir da qual desenvolvedores internos constroem seus novos produtos. A partir do momento que a organização decide por disponibilizar esta plataforma para além dos limites organizacionais, ocorre a transição de linha de produtos de software para ecossistemas de software (BOSCH, 2009). Embora pareça simples, a adoção da abordagem de SECO representa uma mudança significativa no estilo tradicional de desenvolvimento de software e metodologia de processo. Atualmente, diversas empresas estão colocando em prática um novo modelo de processos que tem como pilares o englobamento de diferentes empresas e a disponibilização da arquitetura para o envolvimento de terceiros, obtendo o sucesso como resultado quando estão envolvidas com a abordagem de ecossistemas.

Atualmente, as principais organizações que disponibilizam suas plataformas para terceiros são as empresas de computação móvel, tais como Apple (que, com seu ecossistema do iPhone, será alvo do caso de estudo apresentado no capítulo 4), Nokia e Google (CAMPBELL e AHMED 2010).

Ainda segundo Campbell e Ahmed (2010), o advento dos ecossistemas de software tem feito organizações repensarem sobre suas práticas operacionais, e se envolverem com parceiros externos, abrindo sua plataforma para terceiros para que assim consigam atingir seus objetivos de negócios. Entre os fatores essenciais para criação de um ecossistema estão: o tipo de produto, no caso software, que será trabalhado; a abordagem para abertura de plataforma, e o desenvolvimento das relações com os parceiros.

Antes de iniciar as parcerias com organizações externas, Bosch (2009) afirma que a companhia que resolve dar início ao SECO precisa decidir que tipo de abordagem irá utilizar para fechar essas possíveis parcerias: abordagem dirigida ou não dirigida. Na abordagem dirigida, a organização identifica as áreas específicas de funcionalidades que é incapaz ou não está disposta a desenvolver, mas que ainda assim gostaria de oferecer a seus clientes. Neste caso, a empresa irá selecionar parceiros capazes de fornecer soluções para estas áreas específicas e negocia acordos com estes parceiros. Este tipo de acordo normalmente inclui compartilhamento de receitas, e a companhia parceira passa a obter um acesso mais profundo à plataforma e aos produtos desenvolvidos internamente a partir desta plataforma. A abordagem dirigida permite, portanto, que a empresa expanda rapidamente o escopo de funcionalidades oferecidas a seus clientes, sob baixos riscos e baixo investimento inicial em P&D, à custa do compartilhamento de receita com os seus parceiros. No caso da abordagem não dirigida, a companhia oferece a plataforma como base para o desenvolvimento de aplicações sem restringir quem fará uso dela ou o que será desenvolvido. Os desenvolvedores externos, a partir da solução desenvolvida, podem inclusive se tornar concorrentes um dos outros ou até mesmo da própria organização que disponibilizou a plataforma. Tal concorrência pode ser vista como algo positivo, considerando que é o melhor mecanismo para entregar as melhores soluções aos usuários que, como clientes, irão selecionar as soluções que melhor satisfazem às suas necessidades.

Uma das mudanças essenciais a serem feitas a partir da disponibilização de uma plataforma para parceiros externos é a mudança no foco da estratégia. Caso a organização ainda possua uma estratégia focada no produto, é necessário que o foco passe a ser na plataforma. Diferenças significativas existem entre esses dois tipos de estratégia, e uma delas é que a estratégia focada na plataforma não objetiva obter valor para o cliente de forma direta, mas permite que os próprios desenvolvedores externos construam esse valor. Além disso, de acordo com Bosch e Bosch-Sijtsema (2010), para o surgimento de um SECO, as organizações de desenvolvimento de software envolvidas precisam fazer uma transição de uma abordagem centrada na integração para uma abordagem mais orientada a composição, especialmente devido às dificuldades de coordenação e significativa ineficiência encontradas quando o software como um todo possui um alto acoplamento, gerado pela integração. O uso correto da abordagem composicional é uma importante iniciativa para se alcançar alta qualidade, eficiência e eficácia nas práticas de engenharia de software, e para se obter resultados no desenvolvimento de software em qualquer organização.

A criação de um SECO a partir da disponibilização de um software proprietário é considerada um processo complexo e que pode ser afetado por diversos fatores. Kilamo et al. (2011) enfatizam a importância desses fatores em ecossistemas open source, embora também possam ser considerados para outros tipos de ecossistemas de software, e os agrupa em seis dimensões:

- **Software:** Destaca a importância da qualidade do software disponibilizado, especialmente em termos de usabilidade e de disponibilização de manuais e descrições de arquitetura.
- **Infraestrutura:** Consiste nas ferramentas e tecnologias que viabilizam a comunicação entre os membros da comunidade, a coordenação do projeto, a facilidade de acesso ao mesmo, e a gestão de repositório.
- **Processo:** São necessários processos que equilibrem as práticas da comunidade e as necessidades da empresa no que se refere à tomada de decisões que tenham como objetivos a evolução do software, as ações de manutenção, e o gerenciamento dos *releases*

- **Legalidade:** A empresa que disponibiliza a plataforma deve escolher o tipo de licença para o produto, e deve se responsabilizar pelas questões legais sobre direitos autorais, bem como as questões de marcas e patentes.

- **Marketing:** Estratégias eficazes de marketing são necessárias para o conhecimento da plataforma pelos potenciais usuários e desenvolvedores.

- **Comunidade:** As companhias que estão participando da comunidade devem ser treinadas às novas regras e precisam receber responsabilidades claras. Além disso, o relacionamento entre os *players* deve ser construído com base na confiança mútua.

Para dar início a um ecossistema de software, as organizações tradicionais serão significativamente impactadas pela mudança desse novo contexto. Tais impactos e o quanto isso poderá colocar em risco o desenvolvimento do ecossistema serão melhor detalhados nos próximos capítulos.

3.1. Aspectos que afetam o surgimento e desenvolvimento de um SECO

Como visto, Iansiti e Levien (2004) afirmam que há um paralelo forte entre os ecossistemas biológicos e ecossistemas de negócios. E como visto no capítulo de introdução, esse mesmo paralelo pode ser considerado também para ecossistemas de software. Todos eles são caracterizados por um grande número de participantes, interligados entre si, e que dependem um do outro para sua eficácia e sobrevivência. Devido a essa rede interligada de parceiros, os autores ressaltam que se o ecossistema está saudável, cada participante irá prosperar. Porém, se o ecossistema como um todo não é saudável, seus participantes irão sofrer. Cada parceiro compartilha, portanto, o seu destino com os demais parceiros existentes. Uma empresa, por exemplo, que realiza uma ação sem compreender o impacto em seus domínios de negócios vizinhos, ou sobre o ecossistema como um todo, está ignorando a realidade do ambiente em que opera. Isto faz com que o conceito de saúde se torne de extrema importância para cada *player* envolvido, devendo ser um objeto de estudo e de interesse para todos os membros, para

que, conhecendo este conceito mais a fundo, ele possa então ser utilizado de maneira estratégica dentro da comunidade.

De acordo com Moore (1993), para evoluírem de forma rápida e eficaz, as empresas devem atrair recursos de todos os tipos, com base no capital, em parceiros, fornecedores e cliente para que sejam criadas redes de cooperação. Ele diz que parceiros ajudam a preencher o pacote completo de valor para os clientes. Os ecossistemas e seu desenvolvimento global possuem diversas vantagens, contudo trazem consigo também um conjunto de desafios, tais como diferenças culturais, de fuso horário, nível de maturidade da engenharia de software e diferentes habilidades técnicas entre os diversos parceiros existentes. Além da tecnologia, aspectos gerenciais, sociais e de negócios devem ser considerados como influenciados pelo ecossistema ou influenciadores diretos da sua saúde.

Os estudos mais aprofundados em ecossistemas de software e a tendência de disponibilizar um software proprietário para desenvolvedores externos são relativamente recentes, existindo, portanto, poucas orientações sobre como criar e manter de forma sustentável um ecossistema de software. Como um ecossistema sustentável é sinônimo de ecossistema saudável, é necessário conhecer os aspectos que podem influenciar a saúde de tais ecossistemas ou podem ser influenciados por ele, determinando o seu sucesso ou fracasso. É importante destacar que o grande responsável por promover a saúde global do ecossistema é o *keystone player* e descobrir como este papel pode revitalizar a comunidade e, por consequência, o ecossistema é uma questão importante para manutenção e sobrevivência do mesmo.

Neste trabalho, os principais aspectos influenciados ou influenciadores encontrados foram subdivididos em três grupos: Aspectos Sociais, Aspectos Culturais e Aspectos Geográficos. Estes aspectos estão representados na Figura 3 e serão melhor detalhados nas seções seguintes.

3.1.1. Aspectos Sociais

A análise dos aspectos sociais que influenciam a saúde do SECO engloba os fatores que podem agir sobre o comportamento e características da comunidade envolvida no ecossistema, ou os fatores que podem ser impactados com este tipo de abordagem. Tais aspectos são importantes de serem analisados devido ao valor inerente à comunidade que se torna evidente quando se trata do campo de inovação e desenvolvimento de produtos com sucesso. Tradicionalmente, para analisar e prever como o software evolui, a maioria dos estudos de software depende apenas do código fonte observado. Goeminne e Mens (2010) afirmam que é necessário analisar muitos outros elementos para que se tenha uma visão completa sobre a evolução do software, e destaca, em particular, os aspectos humanos que desempenham um papel significativo de como e por que o software evolui ao longo do tempo.

No que se refere à tomada de decisões do negócio, manter informações sobre a comunidade no geral e sua evolução desempenha um papel fundamental. Para isso, é necessário que dados sejam coletados e analisados, o número de usuários e de parceiros envolvidos deve ser conhecido. A quantidade de vendas, de acesso a sites envolvidos com o sistema, de feedbacks recebidos, a visibilidade do projeto nas redes sociais como Facebook e Twitter, quantidade de pessoas quem participam de eventos e reuniões, número de publicações científicas que citam a comunidade, e a distribuição geográfica dos membros devem ser acompanhados. Em termos de software, a quantidade de downloads feitos, bugs reportados e solicitações de recursos são valiosas. A quantidade e impacto das contribuições recebidas, e seus tipos (corretiva, adaptativa, de aperfeiçoamento, preventiva) também devem ser analisados. Medir e avaliar continuamente a comunidade dá informações valiosas sobre o desenvolvimento do ecossistema, e as tendências observadas nas mudanças de medidas dizem muito sobre o estado da comunidade e apontam possíveis problemas que podem requerer atenção especial (KILAMO et al., 2011).

De acordo com Goeminne e Mens (2010), para garantir o sucesso da produção de um sistema de software, todas as formas de interação humana devem ser levadas em consideração. No âmbito de ecossistemas, para que a expansão do mercado seja bem sucedida, os participantes do SECO precisam criar relacionamentos mais próximos um

com os outros, havendo cooperação entre eles, fazendo com que o mutualismo se dê de forma muito intensa.

Yu et al. (2008) afirmam que alguns biólogos consideram que a simbiose (vista na seção 2.2) é uma importante força motriz para a evolução biológica. Eles acreditam que o conceito de Darwin sobre a seleção natural ser impulsionada pela concorrência está incompleto, e que a seleção é fortemente baseada também em resultados de cooperação e dependência mútua, ou seja, as espécies que são selecionadas pelas mudanças ambientais não são apenas aquelas que foram bem sucedidas numa competição, mas sim aquelas que aprenderam também a cooperar uma com as outras. Tal comportamento pode ser justificado pelo fato de que os *players*, assim como as espécies em ecossistemas biológicos, compartilham seu destino com os demais *players* existentes no ecossistema. Sendo assim, o seu sucesso é determinado, também, pelo sucesso dos outros parceiros, ainda que exista competição entre ambos. Esse tipo de analogia com os ecossistemas biológicos pode, portanto, ser utilizado de forma estratégica para a prosperidade de membros dos SECOs. É importante destacar que não só as relações internas ao ecossistema devem ser levadas em consideração. Além delas, as relações entre diferentes comunidades também devem ser consideradas já que elas podem, de alguma forma, influenciar o seu desenvolvimento, especialmente quando se trata de disputas entre os ecossistemas por uma mesma fatia do mercado.

Dar início a um ecossistema envolve múltiplos atores com diferentes objetivos e até interesses conflitantes. Ao iniciar um ecossistema, a empresa que o fez já deve ter uma ideia dos parceiros que deverão estar envolvidos. Criar um planejamento cuidadoso desses parceiros poderá impactar positivamente a forma como o mesmo será iniciado. Um dos desafios para uma empresa que resolve formar um ecossistema é construir e desenvolver os relacionamentos com as organizações envolvidas. Com o surgimento de um SECO, diferentes *players* dos mais diversos lugares, e com variadas especializações passam a co-existir em uma mesma comunidade. Com isso, complexas relações harmônicas ou desarmônicas (apresentadas na seção 2.2) surgem entre eles, tornando difícil a interação global entre os membros. Para gerir esta interação, é necessário que haja uma mudança do tradicional modelo centralizado de gestão. Embora sejam complexas, as interdependências existentes podem ser gerenciadas e utilizadas de maneira estratégica, levando o ecossistema a uma rede produtiva e inovadora.

Outro aspecto essencial, e que deve ser abordado no contexto de ecossistemas, é a comunicação, que deve ser vista como um tipo investimento a ser feito, que segundo Farbey e Finkelstein (1999), deve ser considerado em diversos níveis: financeiro, pessoal, interorganizacional e estratégico. A comunicação entre os desenvolvedores, clientes e usuários é decisiva já que, a partir dela, requisitos são transmitidos e problemas são relatados. A comunicação entre os parceiros no geral é, também, de igual importância já que, através dela, a evolução do sistema se torna conhecida.

Em termos de contribuição entre os parceiros, segundo Goeminne e Mens (2010), a empresa que dá início ao ecossistema deve estar preparada para apoiar a comunidade como um todo, inclusive fornecendo treinamentos, quando necessário. Além disso, as contribuições de cada parceiro devem ser apresentadas de maneira clara e um nível considerável de confiança deve existir entre eles. Para a manutenção de um ecossistema ativo, é necessário também que o *keystone* cultive a prática de transferência de conhecimento; de resolução colaborativa de problemas; e de compartilhamento de ideias, componentes de software ou configurações (IANSITI e LEVIEN, 2004), contudo, a sua colaboração com os membros deve ser fornecida conforme as suas necessidades e de maneira balanceada. Kilamo et al. (2011) dizem que como um ecossistema lida com diversos parceiros que possuem diferentes interesses, deverá haver equilíbrio também com relação à influência estabelecida entre as partes, para que pelo menos um nível mínimo de satisfação seja garantido entre elas. Além desse equilíbrio, normas sociais e questões jurídicas devem ser consideradas. A não consideração desses aspectos pode comprometer o sucesso da viabilidade do ecossistema. Eles destacam também a importância dos usuários finais como um papel influenciador no que se refere à evolução de um ecossistema, podendo eles fornecer algum tipo de apoio que vai desde testes e feedbacks até uma documentação completa e detalhada do sistema.

Não só um plano de gerenciamento da interação entre os parceiros deve ser desenvolvido, é necessário que em um ecossistema haja também um plano motivacional que atraia potenciais parceiros ao mesmo. Para manter e atrair novos parceiros é importante saber quais eventos influenciam a sua participação. Ververs et al. (2011) elaborou um caso de estudo sobre o Debian, um sistema operacional open source para desktops e servidores, para verificar quais são os pontos que podem atrair desenvolvedores para esta comunidade. Os principais fatores observados foram: eventos

onde diversos parceiros se reúnem para discutir determinados assuntos que envolvem o ecossistema, lançamento de novas versões do software e introdução de novos serviços para os desenvolvedores. Jansen et al. (2009) listam outros principais motivos que podem manter e atrair novos parceiros, são eles: compartilhamento de receita, introdução de recompensas diretas, estabelecimento de uma rede de parceiros e programas de certificação. Eles dão o exemplo da organização GX, empresa holandesa que trabalha no mercado de sistemas de gestão de conteúdo, que atraiu *players* a partir da criação de um ambiente *partner-friendly*, de seu produto bem sucedido, e da criação de novas oportunidades de negócios pela conscientização de que uma rede de parceiros é necessária para poder satisfazer todos os seus clientes. Esse tipo de característica pode identificar externamente um SECO dos demais, fornecendo às organizações uma visão rápida dos limites que o definem e suas principais propriedades para uma análise das oportunidades e ameaças potenciais existentes. Esses autores dizem que um observador externo que tem interesse em participar do ecossistema terá interesse em saber qual a organização *keystone*, quem são os fornecedores, quais são as normas que regem o ecossistema, ou seja, quais as informações que definem como o SECO age e se desenvolve. Além disso, um observador externo vai querer saber quais os tipos de clientes que estão ativos no SECO, e a ligação que esse ecossistema tem com outros tipos de ecossistemas, pois segundo Jansen et al (2009), se um SECO estiver estreitamente relacionado com diversos outros SECOS, seus participantes irão se recuperar melhor de mudanças radicais que possam surgir. Um ecossistema se torna mais interessante quando se observa sua performance, o valor de seus desenvolvimentos mais recentes, a quantidade de participantes e a quantidade de clientes existentes.

Campbell e Ahmed (2010) definem uma dimensão social que aborda três razões motivacionais pelos quais novos parceiros são atraídos a fazerem parte do ecossistema:

- **Utilitarismo:** Parceiros atraídos por esta característica são motivados por fatores utilitários, e buscam algum tipo de recompensa pelo seu engajamento no ecossistema. Tal recompensa poderá ser sob pagamento monetário ou não.

- **Promoção:** A promoção pode ser implícita ou explícita. Parceiros do SECO que são atraídos pela promoção implícita não estão ativamente à procura de reconhecimento ou recompensa pelo seu engajamento. Eles estão engajados por intenções altruístas ou

simplesmente pelo desejo de se envolver com a tecnologia em si. A promoção explícita está intimamente ligada a fatores utilitários, nos quais os colaboradores são movidos pelo desejo de buscar reconhecimento pelo seu trabalho e, assim, poderão ganhar algum tipo de recompensa.

- **Ganho de Conhecimento:** O desenvolvimento de ecossistemas de software criou novas oportunidades no que se refere ao engajamento das partes interessadas que querem se envolver com novas tecnologias e ferramentas. Esse engajamento permite que desenvolvedores se tornem parte de uma determinada comunidade, e desenvolvam suas habilidades e conhecimentos por meio da contribuição.

É importante ressaltar, que todas as recompensas são controladas pelo *keystone*. Campbell e Ahmed (2010) afirmam que o sucesso do ecossistema, em última análise, depende do quão satisfatoriamente a empresa *keystone* lida com esta dimensão social. Para atrair e reter os membros, é necessário que este tipo de organização gere valor dentro do ecossistema, e compartilhe este valor com os demais participantes de forma balanceada. Iansiti e Levien (2004) afirmam que conseguir esse balanceamento não é tão fácil como parece. Organizações *keystone* devem ter certeza que o valor de suas plataformas dividido pelos custos de criação, manutenção e compartilhamento, aumente rapidamente com o número de membros do ecossistema que fazem uso dela. Isso permite que elas compartilhem o saldo positivo com a sua comunidade. Eles citam a companhia eBay como um bom exemplo de *keystone*, afirmando que ela criou valor para seu ecossistema de várias formas, e entre elas está o desenvolvimento de ferramentas que aumentaram a produtividade dos membros da rede e encorajou novas companhias a se juntarem ao ecossistema. Além disso, a eBay estabeleceu e manteve padrões de performance que melhoraram a estabilidade do sistema. Pelo compartilhamento de valor, ela continua a expandir a saúde do ecossistema, fazendo com que seus parceiros prosperem de forma sustentável.

Keystones também podem aumentar a produtividade simplificando a complexa tarefa de conectar os participantes da rede uns aos outros, e criando através de terceiros novos produtos mais eficientes. Eles podem melhorar a robustez incorporando novas tecnologias, e podem fornecer um ponto de referência confiável aos parceiros,

ajudando-os a responderem a novas e incertas condições (IANSITI e LEVIEN, 2004). Segundo esses autores, *keystones* podem encorajar a criação de nicho não só incorporando novas tecnologias, mas também oferecendo estas inovações tecnológicas às partes envolvidas no ecossistema.

3.1.2. Aspectos Gerenciais

Como visto na seção acima, a comunicação entre os parceiros é essencial para o desenvolvimento dos ecossistemas. A importância da comunicação se dá especialmente quando se trata da gerência dos grupos pertencentes à comunidade, e a gerência do ecossistema como um todo. Na abordagem de ecossistemas de software, não só a evolução da plataforma utilizada, dos produtos desenvolvidos e as equipes dentro da empresa precisam ser coordenados, os desenvolvedores e equipes externas também devem estar envolvidos na coordenação devido à sua ligação direta com o SECO. De acordo com Bosch e Bosch-Sijtsema (2010) muitas vezes esses desenvolvedores possuem grande influência no que diz respeito às prioridades e seqüenciamento das funcionalidades da plataforma.

Durante o desenvolvimento, a arquitetura, mais especificamente as interfaces entre a plataforma e os produtos, deve ser desenvolvida em colaboração com a comunidade de desenvolvedores externos, e não simplesmente publicada. A comunidade de desenvolvedores externos precisa estar envolvida inclusive na validação de uma nova versão para a plataforma. Portanto, a coordenação deve levar em consideração todos esses parceiros externos e ainda deve estar envolvida em todas as fases do ciclo de vida do ecossistema, garantindo inclusive que a incorporação de funcionalidades na plataforma por parte de terceiros não seja contenciosa e nem cause má reputação para a organização que disponibilizou a plataforma (BOSCH, 2009) ou para o ecossistema como um todo.

De acordo com Bosch e Bosch-Sijtsema (2010), sob uma perspectiva organizacional, equipes geograficamente dispersas que desenvolvem componentes que fazem parte de um mesmo sistema ou família de sistemas têm dificuldade significativamente maior na execução da coordenação necessária, acarretando em altos

custos que podem afetar de maneira significativa os benefícios que são normalmente associados a esta abordagem global. Segundo Bosch (2009), a complexidade de coordenação é de tal dimensão que se não for definida uma estratégia eficaz para a mesma, a vantagem competitiva da abordagem de SECO é questionada. Na abordagem de ecossistemas de negócios, Iansiti e Levien (2004) citam que mais da metade dos benefícios de custo no ramo de supermercados de varejos da Wal-Mart resulta de como a companhia gerencia seu ecossistema de parceiros.

Algumas características inerentes aos ecossistemas podem dificultar inclusive a gerência de requisitos que somente é eficaz quando há um mútuo entendimento entre as partes para que não haja qualquer equívoco no conteúdo do contrato, que é de natureza crítica. Para evitar este tipo de equívoco, uma linguagem comum entre as partes é exigida (FARBEY e FINKELSTEIN, 1999). Portanto, é necessário que parceiros pertencentes a países diferentes, com linguagens diferentes, adotem uma linguagem padrão e se abstenham de peculiaridades culturais. Para ecossistemas de software, a comunicação entre as partes interessadas requer, portanto, um padrão formal que modele tanto o ecossistema como o ambiente no qual os produtos de software e serviços operam.

O fato de haver *players*, em especial engenheiros de software, localizados em diferentes países acarreta implicações significativas na comunicação e colaboração das equipes, dificultando conseqüentemente o gerenciamento das mesmas. Bosch e Bosch-Sijtsema (2010) fizeram uma análise sobre diferentes grupos distribuídos geograficamente, trabalhando para um mesmo objetivo, e verificaram que a distribuição das equipes resultou numa baixa na produtividade devido ao *overhead* nas comunicações entre os membros, às diferenças culturais, e à desmotivação das equipes que se encontravam em locais remotos. Outro obstáculo enfrentado por equipes localizadas remotamente é a diferença de fusos horários que acaba por atrasar o envio e recebimento de informações e a comunicação no geral.

Em síntese, trabalhar com equipes remotamente distribuídas aumenta a quantidade de tempo necessária para realizar determinadas tarefas devido a diferenças culturais, de fuso horário, e do gasto de tempo necessário para coordenação do trabalho que engloba diferentes países. Para minimizar tais problemas, é essencial que haja um maior

investimento na coordenação das equipes, porém, tal investimento pode acarretar num significativo impacto negativo sobre a eficiência do desenvolvimento do sistema. Ainda devido à diferença entre fusos horários, certas equipes ou gerentes precisam mudar o seu horário de trabalho, o que pode impactar o aspecto motivacional e, também, tornar o desenvolvimento menos eficiente.

As diferenças culturais e de linguagem são inerentes a globalização e podem acarretar em dificuldades na coordenação dos grupos e também na comunicação eficaz. Devido aos problemas de gerenciamento que foram citados, Bosch e Bosch-Sijtsema (2010) sugerem que as organizações em vez de adotarem uma abordagem centrada na integração, adotem uma abordagem centrada na arquitetura, mais composicional. Para que se possa obter a dissociação necessária para este fim, é importante que a própria arquitetura do software seja revista. Uma das implicações de uma abordagem descentralizada é que a coordenação em todo o sistema não é mais conseguida através de processos de desenvolvimento, mas da própria arquitetura de software estabelecida (BOSCH, 2009), e cada equipe local aumenta consideravelmente o nível de autonomia e responsabilidade.

Muitas vezes, a composição se dá pelo rompimento de dependências desnecessárias entre os componentes e das equipes responsáveis por eles. Bosch e Bosch-Sijtsema (2010) defendem que cada dependência exige um esforço considerável para controlá-la, e a complexidade global do sistema cresce exponencialmente com o crescimento das dependências, o que traz como consequência a redução da produtividade, e o impacto negativo na eficiência global do produto e na plataforma de desenvolvimento. Exemplos de problemas que podem ocorrer quando não se tem uma arquitetura suficientemente dissociada incluem o efeito cascata gerado quando uma equipe atrasa uma determinada entrega, levando ao atraso de outras equipes que trabalham juntas no desenvolvimento do software; e os altos custos de integração do software, que levam obstáculos à evolução do mesmo. Esses autores dizem que embora parte desses problemas possam ser consequências dos processos existentes, a ineficiência é claramente causada pela arquitetura de software definida.

Quando se trata da arquitetura integrada de componentes e equipes, Bosch e Bosch-Sijtsema (2010) citam alguns problemas que podem ser observados nos ecossistemas que seguem essa abordagem:

- **Maior nível de complexidade:** Como falado anteriormente, uma arquitetura integrada de componentes e equipes gera o aumento da complexidade do sistema e de sua coordenação. Para evitar este tipo de problema, têm-se a dissociação da arquitetura como uma possível solução. O objetivo da decomposição do sistema em partes consistentes é que essas partes podem evoluir com relativa independência, minimizando a quantidade de relacionamentos existentes entre elas e conseqüentemente reduzindo a complexidade do sistema. Atingir e manter uma arquitetura simples deve ser prioridade tanto no projeto inicial do sistema, como durante a sua evolução.

- **Obstáculos na evolução do software:** Quando os componentes e equipes são totalmente dependentes, pode ocorrer o fenômeno de que estes componentes só poderão ser laçados, ou seguir para a próxima iteração, simultaneamente, caso contrário o sistema quebra. Isso traz como conseqüência o alongamento no tempo do ciclo de vida do projeto ou produto, podendo fazer com o que o ecossistema perca a vantagem competitiva sobre os demais concorrentes. Este problema pode ser justificado pelo fato de que tais sistemas são fortemente acoplados, ou seja, os dados e a lógica de negócios não estão adequadamente separados da interface do usuário e do fluxo da aplicação.

Podemos, portanto, concluir que para melhorar a gerência e coordenação das diversas equipes ou parceiros existentes no SECO, é necessário minimizar as interdependências entre eles. Uma das soluções para isso é modificar a arquitetura do sistema, fazendo com que cada equipe se responsabilize por componentes que devem estar fracamente acoplados. Isto se dá porque a arquitetura proporciona uma formalização das regras de interoperabilidade, fazendo com que seja possível que as equipes operem de maneira independente (BOSCH, 2009). Uma das sugestões fornecidas por Bosch e Bosch-Sijtsema (2010) para quebrar o sistema em grupos de componentes é utilizar técnicas de dissociação tais como SOA (Service Oriented Architecture), nos quais os componentes subdivididos exigem uma elevada integração quando se trata do contexto dentro do domínio, enquanto que uma integração

significativamente menor é necessária no trabalho entre estes componentes ou domínios. SOA é uma abordagem eficaz na diminuição de ineficiências associadas com a coordenação de equipes localizadas em diferentes lugares, com diferentes fusos horários. Para que essa composição se torne eficaz, é necessária não só a mudança na arquitetura do sistema, mas também uma mudança nos processos existentes dentro do ecossistema. Esta mudança de processos que é necessária será melhor abordada na seção seguinte.

Embora existam diversos benefícios associados à abordagem composicional e aos ecossistemas de software, Bosch (2009) cita que algumas questões ainda ficam em aberto com relação à manutenção de uma experiência atraente com o usuário. Como o sistema como um todo é construído por diferentes partes, a experiência geral do usuário pode ser influenciada, resultando numa oferta menos atraente para o cliente. Embora ainda não exista uma solução que resolva este desafio completamente, as organizações envolvidas, especialmente o *keystone*, devem considerar esta problemática, fornecendo algum tipo de apoio para minimizá-la.

3.1.3. Aspectos de Negócios

Ao dar início a um ecossistema, existem diversos custos associados que devem ser levados em consideração. Como exemplos, têm-se os custos para abertura de plataforma, criação de interfaces externas, restrições da evolução dessas interfaces, além dos desafios de gerência e coordenação do grupo de parceiros envolvidos. Weiss (2011) destaca que criar um ecossistema de software, a partir da disponibilização de uma plataforma, aumenta o esforço de desenvolvimento de software de 1,5 a 3 vezes. Ele afirma ainda que apenas grandes empresas foram capazes de justificar esse tipo de investimento.

A opção de construir ativos reutilizáveis (plataformas) não está disponível para pequenas empresas devido a pressões de tempo e recursos em que operam, além da gestão do fluxo de caixa exigido. Adicionalmente, a transição entre um desenvolvimento de software interno e um ecossistema de software acarreta a necessidade de mudança e reestruturação dos processos que abrangem o sistema

envolvido. Bosch (2009) diz que a principal mudança na transição entre essas abordagens é que as equipes externas não podem ser submetidas a padrões de modelos de processos, ferramentas e formas de trabalho tradicionais, o que significa que modelos de referência que contêm práticas necessárias à maturidade dos processos, como CMMI, tornam-se mais difíceis de serem aplicadas neste tipo de abordagem. A mudança de processos não somente acontece em termos de processos relacionados ao desenvolvimento de software, eles abordam, inclusive, a gestão de requisitos e a interação entre os parceiros. Isto significa que os processos otimizados voltados para o ambiente intra-organizacional não são mais úteis, ou caso ainda sejam úteis, não são eficazes, concluindo-se que alternativas devem ser encontradas.

O desenvolvimento tradicional de software tende a se basear fortemente em mecanismos centralizados, em termos de gerenciamento de requisitos, evolução de arquitetura, integração e garantia de qualidade. Bosch (2009) ressalta que a solução em alto nível para este desafio é passar de um sistema centralizado para uma abordagem descentralizada, descentralizando então as atividades envolvidas nesses processos. É, portanto, a transição para a abordagem composicional explicada na seção anterior. O maior obstáculo no uso de processos tradicionais é que normalmente eles dependem de uma comunicação face a face entre os membros, ou assumem que haverá uma interação entre eles durante o desenvolvimento de uma iteração, causando significativa ineficiência na produtividade de organizações globais (Bosch e Bosch-Sijtsema, 2010).

No geral, os processos devem balancear as práticas da comunidade e as necessidades das companhias envolvidas quando se trata da tomada de decisões para a evolução do software em questão, das ações de manutenção, além do gerenciamento dos *releases*. A governança tanto dos projetos como das relações existentes no ecossistema deve fazer parte do conjunto de processos existentes. Dentre as mudanças envolvidas na transição de processos centralizados para processos descentralizados se incluem também o papel da gestão de configuração de software e a garantia de qualidade, que passam a não mais enxergar a funcionalidade global do sistema, enxergando apenas as funcionalidades de cada componente em específico. A verificação descentralizada da qualidade permite a minimização da quantidade de esforço que precisa ser gasto no enfoque global (BOSCH, 2009).

Bosch e Bosch-Sijtsema (2010) defendem que a arquitetura de software é central na dissociação do sistema, permitindo uma abordagem composicional, mas a considera apenas um facilitador para se chegar a tal objetivo. Eles dizem que a composição real do sistema é comprometida pela implementação de processos. Sendo formais ou informais, os processos de engenharia definem a colaboração concreta entre equipes e entre indivíduos. Baseado em alguns casos de estudo, esses autores listam alguns problemas na engenharia de processos que devem ser evitados na abordagem composicional, e que, portanto, devem ser evitadas no domínio dos ecossistemas de software. Alguns dos problemas citados são:

- **Características não especificadas que cruzam os limites de um componente:** São características relacionadas a dois ou mais componentes que não foram especificadas antes do ciclo de desenvolvimento ser iniciado e começam a ser trabalhadas durante o desenvolvimento. Isso é considerado um problema porque, na prática, requer que haja uma estreita interação entre as equipes envolvidas, causando um significativo *overhead* que poderia ser facilmente evitado durante a etapa anterior de especificação.

- **Integração feita apenas no final do ciclo de desenvolvimento:** A integração por si só já requer certo nível de complexidade e demanda altos custos para ser feita devido à incompatibilidade existente entre os componentes. Quando feita no final do ciclo de desenvolvimento, ela pode se tornar ainda mais dispendiosa e imprevisível, causando dificuldades significativas às empresas envolvidas.

- **A falta de disciplina nos processos:** É um problema que pode afetar qualquer tipo de abordagem, sendo ela composicional ou não. Embora existam processos coerentes com a abordagem de ecossistemas, é possível que alguns membros resolvam ignorá-las. Bosch e Bosch-Sijtsema (2010) exemplificam o caso de um engenheiro de software durante um desenvolvimento que cria uma ligação entre dois ou mais componentes distintos, embora saiba que a arquitetura utilizada não permita quaisquer ligações desse tipo. Como resultado disso, a validação independente dos componentes torna-se impossível de se fazer. A integridade da arquitetura foi violada, e durante a evolução do sistema essas dependências implícitas podem causar ineficiências no processo de desenvolvimento.

- **Processos não estão adequados ao tipo de abordagem utilizada:** Embora os processos sejam definidos e seguidos na organização, eles devem estar alinhados ao tipo de abordagem utilizada. Como falado anteriormente, no caso da abordagem de ecossistemas de software, os processos desenvolvidos não devem depender, em sua totalidade, de uma comunicação presencial entre as equipes envolvidas, e não devem assumir que haja interação frequente entre as mesmas, especialmente pelo fato de que elas podem estar localizadas a uma distância geográfica considerável, e com grandes diferenças de fusos horários.

- **Processos de Negócio e de Engenharia desalinhados:** Embora os processos de engenharia estejam alinhados com a abordagem global, os processos de negócios podem não estar adequados a este tipo de abordagem. Quando uma nova estratégia de negócio é desenvolvida, toda a organização deve mudar de forma significativa, não somente o setor voltado para pesquisa e desenvolvimento. Caso a organização como um todo não evolua quando ocorre a mudança de abordagem, a administração do produto a ser desenvolvido pode não ser capaz de fornecer a orientação necessária para o desenvolvimento do mesmo, podendo acarretar em perdas de inovação e de oportunidades de negócio.

Outro aspecto da área de negócio que influencia o desenvolvimento de um ecossistema é o conhecimento tácito inerente a cada organização. Tal conhecimento é implícito e enraizado, e é considerado um aspecto importante quando se trata do trabalho de equipes globais. Práticas de trabalho, conscientização das diferenças culturais, bem como a consciência contextual de um local podem ser definidos como conhecimento tácito dos membros de uma equipe local, que é difícil de ser compartilhado ou transferido para equipes remotas (BOSCH e BOSCH-SIJTSEMA, 2010). Esse tipo de conhecimento deve ser levado em consideração nos processos de mudança organizacional, já que quando uma nova estratégia de negócio é estabelecida, parte deste conhecimento precisa ser mudada, embora seja difícil, já que é um conhecimento implícito. Devido a esta dificuldade de mudança, pode haver certa resistência por parte das equipes locais quando se trata da mudança de processos e estratégias organizacionais. Por consequência, essas resistências podem causar

ineficiências na transição de abordagem, demandando um esforço significativo por parte da organização para tentar modificar de forma geral o comportamento das equipes envolvidas.

A figura abaixo apresenta o modelo agrupado dos aspectos que foram abordados neste capítulo. Cada grupo de aspectos representa um conjunto de fatores que devem ser levados em consideração quando organizações decidem fazer parte de um ecossistema de software, devido à influência que cada um deles pode exercer no desenvolvimento do SECO.

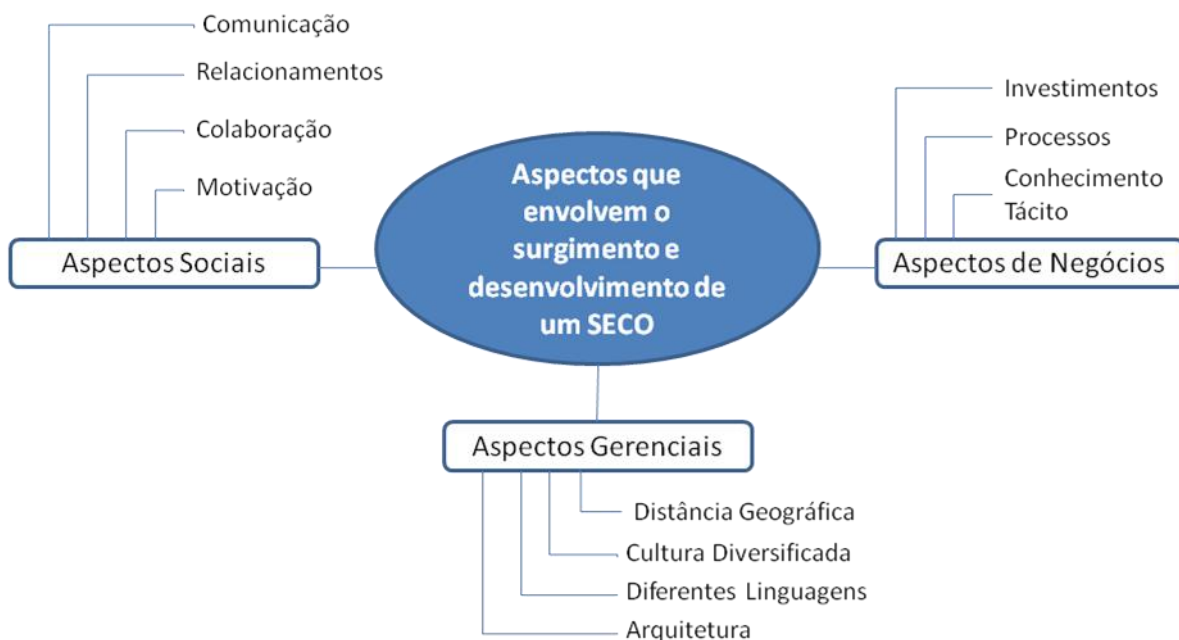


Figura 3. Aspectos que envolvem o surgimento e o desenvolvimento de um SECO.

4. Ecossistema do iPhone

Embora o Ecossistema de Software do iPhone tenha nascido em 1999, quando a Apple registrou o domínio iPhone.org (ANDRADE et al., 2011), o iPhone foi anunciado ao público somente em Janeiro de 2007 por Steve Jobs. Este ecossistema incorporou parte do ecossistema do iPod, que foi lançado em 2001 e tornou-se bastante popular, especialmente após o lançamento do *iTunes Music Store* em 2003, quando mais de um milhão de iPods foram vendidos.

A estratégia para iniciar um SECO depende de como o *keystone* convence um *niche player* a adotar a sua tecnologia. Devido à popularidade do iPod, à forte reputação entre os seus usuários, à qualidade e inovação de seus produtos, muitas organizações despertaram o interesse em cooperar com a Apple em seu novo ecossistema, especialmente empresas que prestam serviços em telecomunicações. Atualmente, *players* de diversos segmentos atuam dentro do SECO do iPhone, em busca do seu sucesso, para que lucros e benefícios sejam compartilhados.

Segundo Andrade (2012), este ecossistema, que possui rendimentos elevados, está envolvido por uma plataforma robusta composta por sistema operacional, kit de desenvolvimento de software (SDK), loja on line, numerosos *niche players* espalhados pelo mundo inteiro e diversas aplicações que são beneficiadas pela plataforma. O iPhone como ecossistema é gerado a partir de um esforço global. Mais de 30 organizações em 3 continentes estavam envolvidas no lançamento do primeiro iPhone (MATTHEW, 2009). Esse autor apresenta uma lista parcial dos primeiros parceiros envolvidos no SECO do iPhone que é apresentada no Quadro 2. Algumas das responsabilidades da Apple foram, e ainda continuam sendo, o design do produto e seu software. Atualmente, diversas empresas contribuem para o desenvolvimento do SECO do iPhone, cada uma contribuindo com partes complexas de seu projeto. Muitos desses *players* são bastante conhecidos, tais como *US-based Intel*, *Korea's Samsung*, *Japan's Sharp* e *Sanyo Epson*.

Primeiros parceiros do SECO do iPhone		
Software e design	Apple	EUA
Assembly	Foxconn	Taiwan
TFT-LCD Screen	Sanyo Epson, Sharp, TMD	Japão
Video processor chip	Samsung	Coreia
Touch screen overlay	Balda	Alemanha
Bluetooth chip	Cambridge, Silicon Radio	Reino Unido
Chip manufacture	TSMC, UMC	Taiwan
Baseband IC	Infineon Technology	Alemanha
WIFI chip	Marvell	EUA
Touch screen control chip	Broadcom	EUA
CMOS chip	Micron	EUA
NOR Flash ICs	Intel, SST	EUA
Display Driver chip	National Semi, Novatek	EUA
Case Mechanical parts	Cacther, Foxconn Tech	Taiwan
Camera lens	Largan Precision	Taiwan
Camera module	Altus – Tech, Primax, Lite On	Taiwan
Battery Charger	Delta Eletronics	Taiwan
Timing Crystal	TXC	Taiwan
Passive components	Cyntec	Taiwan
Connector and cables	Cheng Uei, Entery	Taiwan

Quadro 2. Os primeiros parceiros do ecossistema do iPhone.

Como visto na tabela acima, os players participantes do SECO do iPhone se localizam em diferentes lugares do mundo, com diferentes idiomas locais, e até mesmo fusos horários opostos. Relações complexas envolvem este ecossistema, como exemplo tem-se a Samsung, que é parceira da Apple neste ecossistema, e ao mesmo tempo é sua maior concorrente em relação aos tablets, e aos próprios smartphones. Apesar de

disputas judiciais referentes às patentes de produtos estarem ocorrendo entre estas duas empresas desde abril de 2011, elas ainda mantêm a parceria, e a Samsung permanece fornecendo seus chips para a construção de iPhones.

A incorporação de desenvolvedores externos independentes ao ecossistema do iPhone se deu em 2008, após a Apple lançar o SDK do iPhone (*Software Development Kit*), permitindo que desenvolvedores criassem aplicativos para o iPhone e iPod touch e ainda executassem o simulador do iPhone no Mac. A partir daí, milhares de desenvolvedores passaram a cooperar com o ecossistema através do desenvolvimento de aplicativos que são disponibilizados na *Apple Store*, aumentando consideravelmente o *niche player* do ecossistema.

Ainda que a plataforma de desenvolvimento de aplicativos tenha sido disponibilizada para terceiros, muitas limitações ainda são impostas para o seu uso. Além da restrição de pagamento pelo registro no *iPhone Developer Program*, a Apple impôs a necessidade de se utilizar o Mac para desenvolvimento de aplicativos. A companhia disponibilizou o *Xcode* (Figura 5), IDE para desenvolvimento de aplicativos que devem utilizar a linguagem *Objective-C*. Embora os desenvolvedores possam fazer uso de uma cópia gratuita desse ambiente de desenvolvimento para a construção de aplicativos, eles precisam pagar uma licença anual de \$99.00 para que possam lançar seus produtos na *Apple Store* e obter algum tipo de lucro. Mesmo com essas restrições, em 2009 a Apple já contava com 125.000 desenvolvedores para iPhone registrados no *iPhone Developer Program*, e este número vem crescendo constantemente (BROOKS, 2009).

A incorporação de diversos desenvolvedores de aplicativos ao ecossistema, a partir da disponibilização da primeira versão beta do SDK, foi consequência da demanda por este trabalho ter disparado em 500% em seis meses, desde abril de 2008, segundo a oDesk, mercado de terceirização líder em empregos de tecnologia (COOPER, 2008). Este aumento é apresentado na Figura 4. O resultado desta demanda é o número de 377.000 aplicativos disponíveis na *Apple Store* do iPhone em 2011 (IDU et al., 2011), e ainda em 2010, 18 meses após o lançamento da *Apple Store*, mais de 4 bilhões de aplicativos foram baixados (BERGVALL-KAREBORN, 2010).



Figura 4. Demanda de desenvolvimento de aplicativos para iPhone.

Com relação ao compartilhamento da receita, o desenvolvedor externo pode definir qualquer preço para o seu aplicativo desde que esteja acima de um limite mínimo imposto pela Apple, que receberá 30% do valor adquirido pelo desenvolvedor com as vendas desses aplicativos. Uma alternativa a isso é a disponibilização gratuita do aplicativo pelo desenvolvedor, não ficando este responsável por pagar quaisquer custos com sua liberação e distribuição, exceto a taxa de adesão.

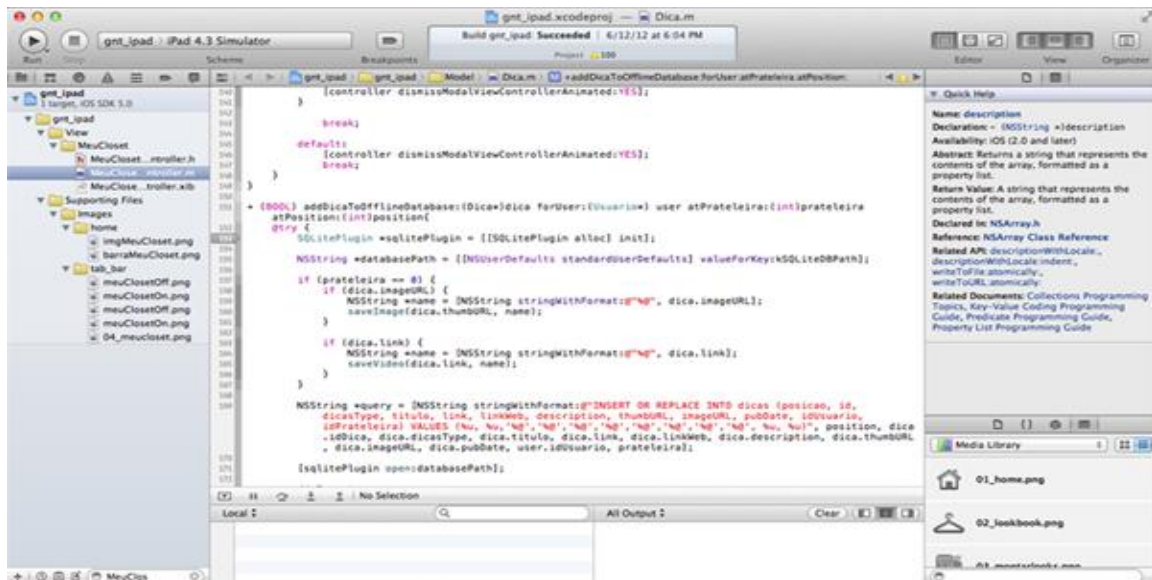


Figura 5. IDE para desenvolvimento de aplicativos.

Andrade (2012) reúne os principais parceiros deste ecossistema em 4 grupos:

- **Apple:** Organização *keystone* do ecossistema. É a empresa responsável pela plataforma de desenvolvimento, que é composta pelo sistema operacional (iOS), a Apple Store e o kit de desenvolvimento SDK;
- **Fornecedores de dispositivos:** Como falado anteriormente, a Apple não fabrica o iPhone. Exemplos de fornecedores estão apresentados no Quadro 2, e os tipos de acordos feitos entre eles são, em sua maioria, confidenciais;
- **Cadeias de varejo e operadoras de telecomunicações:** Como a Apple não vende o iPhone diretamente, ela define acordos comerciais com redes de varejo e com as operadoras de telecomunicações para adquirir seus clientes;
- **Desenvolvedores independentes:** São os que desenvolvem os aplicativos para o iPhone. Em termos de quantidade, eles são maioria neste ecossistema.

Ainda segundo Andrade (2012), a Apple exerce total controle sobre o SECO do iPhone, inclusive na aprovação da publicação dos aplicativos na Apple Store. O seu modelo de negócio é baseado especialmente nas vendas dos dispositivos e download de aplicativos. O autor mostra uma visão geral do SECO do iPhone através da Figura 6.

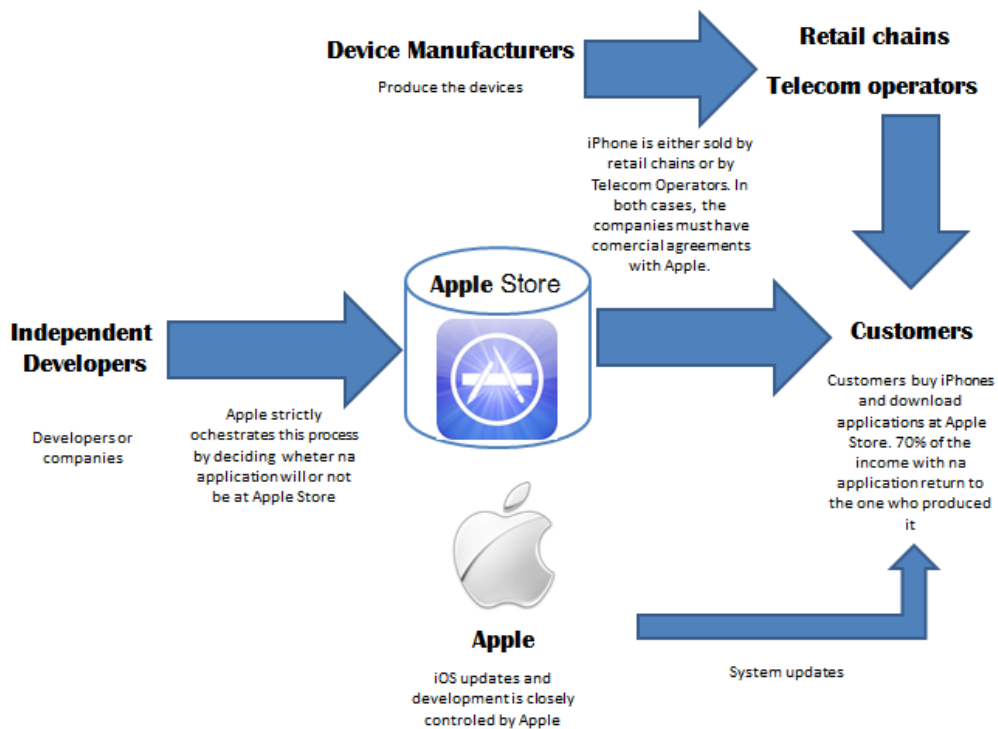


Figura 6. Visão geral do ecossistema do iPhone.

Iansiti e Levien (2004) afirmam que um *keystone* pode melhorar a robustez de um ecossistema incorporando inovações tecnológicas. A Apple executa esse papel no ecossistema, como afirmado por Bergvall-Kareborn (2010), que afirma que segundo os desenvolvedores que fazem parte desse ecossistema, o fato que os motiva a desenvolver aplicativos para o iPhone é sua tecnologia ser a mais avançada, beneficiando, portanto, as aplicações desenvolvidas. Outra motivação citada é o fato de haver poucas barreiras para o desenvolvimento, na forma de tempo, recursos e facilidade de aprendizado, quando comparada com outras plataformas. Além disso, de maneira geral, a reputação do *keystone* influencia significativamente em termos de atrair e manter parcerias existentes e, nesse aspecto a Apple desempenha bem a sua função.

Como falado anteriormente, a popularidade do iPod, a forte reputação entre os seus usuários, a qualidade e inovação de seus produtos despertam o interesse das organizações e dos desenvolvedores independentes, não só por questões monetárias, mas também, por questão de status. Ainda, para que um *keystone* consiga atrair ou reter seus membros, é necessário não só a criação de valor para o ecossistema, mas também o

compartilhamento desse valor com os demais membros. A Apple já vem executando essa ação como *keytone*, especialmente no compartilhamento de receitas, como foi visto. Além disso, ela faz o seu papel contribuindo com o ecossistema no que se refere, por exemplo, a disponibilização de diversos recursos do *IOS Dev Center*, tais como vídeos, amostras de códigos, documentação técnica e fóruns, que colaboram para o aprendizado no desenvolvimento de aplicativos, especialmente para iniciantes. O *Developer Program* fornece aos desenvolvedores acesso aos pré-lançamentos de software do iPhone, suporte técnico, e capacidade de se obter código para testes no próprio iPhone (em vez de se usar o simulador no Mac).

Em termos de relacionamentos entre a Apple e seus parceiros, dos limites geográficos e dos diferentes fusos horários, pouco se sabe sobre como a Apple lida e coordena os diversos parceiros nesta situação, mas é notável a independência e a auto-suficiência existente em cada um deles, bem como a divisão clara das responsabilidades que cada player possui. Com relação à linguagem, o padrão utilizado neste ecossistema é o Inglês, o que minimiza os obstáculos das diferentes localizações a que os *niche players* pertencem.

A estabelecida experiência da Apple com produtos como o Mac e iPod juntamente com seu talento de criar produtos exclusivos e com design inovador, e a cooperação recíproca existente dentro do SECO do iPhone tornaram esta organização a líder mundial da telefonia celular em termos de faturamento, desde o primeiro trimestre de 2011 (MENDES, 2011). Segundo Andrade (2012) em termos de estratégia, a Apple conseguiu abordá-la de forma satisfatória durante todo o ciclo de vida do SECO do iPhone, resultando em ganhos e aprendizados coletivos dentro do ecossistema. Contudo, mesmo obtendo todo este sucesso, a Apple deve continuar a fazer uso de estratégias de forma sábia para que o fortalecimento de ecossistemas concorrentes como Android e Windows Mobile não afete a sustentabilidade alcançada até agora.

5. Conclusões e Trabalhos Futuros

É notável que sistemas de software não funcionam de forma independente. Como esta pesquisa demonstrou, as aplicações de software normalmente trabalham em conjunto com sistemas operacionais, sistemas de banco de dados e/ou sistemas de redes, não só dependendo de elementos computacionais, mas também de fatores externos ao ambiente de software, tais como clientes, usuários, e mercados. Projetos de software são, portanto, rebuscados e complexos. Pesquisas na área de SECO's vêm se mostrando bastante úteis dentro desse contexto. Um ecossistema de software é, em seu sentido mais fundamental, um sistema no qual as paredes tradicionais entre as entidades de desenvolvimento são quebradas, permitindo que haja colaboração e interoperabilidade entre diversas organizações diferentes. O conceito de ecossistemas de software refere-se à forma como as empresas interagem para criar e entregar os produtos, de forma que cada parceiro, dentro de uma rede interligada de empresas, tem um papel e desempenho individual que influenciam a saúde do ecossistema como um todo.

SECO's estão cada vez mais ganhando popularidade entre as grandes organizações devido aos diversos benefícios trazidos por este conceito, tais como compartilhamento de custos e aumento de produtividade, o que leva a um significativo diferencial em termos de competitividade. As empresas são motivadas a tomarem esse tipo de decisão pelo diferencial que elas podem alcançar quando comparada a seus concorrentes, já que estatísticas mostram que na abordagem tradicional entre 50% e 90% do esforço de desenvolvimento de software é gasto com a criação de software que não traz qualquer diferencial a empresa que o desenvolveu. Sendo assim, a disponibilização de uma plataforma contribui para que as organizações apenas invistam tempo para desenvolver produtos especializados que lhes trarão algum tipo de diferencial quando comparada com as demais organizações.

Em ecossistemas de software, muitas características são provenientes dos ecossistemas biológicos. Os conceitos ecológicos de co-evolução, auto-organização e emergência são trazidos para os SECOs, bem como as relações harmônicas e desarmonicas existentes entre os diversos indivíduos de espécies diferentes. Um conceito herdado da biologia de significativa importância é o de saúde do ecossistema,

que é forma que se tem de avaliar a sua força em um momento específico. Para avaliar essa força, é necessário que perspectivas como robustez, produtividade e criação de nicho sejam medidas. Cada parceiro pertencente à rede possui um papel definido dentro do ecossistema. Dois deles podem ser reproduzidos com segurança pelas organizações: *keystone* e *niche player*.

As empresas optam por disponibilizar sua plataforma para terceiros, entre outros motivos, devido ao aumento da oferta básica para os usuários existentes, à capacidade de atrair novos usuários, à aceleração da inovação e ao compartilhamento de custos. Quando uma organização decide disponibilizar seu software e dar início a um ecossistema, ela deve considerar diversos aspectos advindos da mudança da abordagem tradicional da organização para uma abordagem global que envolve parceiros externos. É necessário identificar e analisar tais aspectos porque, se não considerados, eles podem impactar a saúde do ecossistema, e fazer com que os benefícios trazidos por essa abordagem sejam comprometidos, levando ao não desenvolvimento do mesmo e conseqüentemente a sua morte. Para melhor abordar esses aspectos, este trabalho dividiu-os em três grupos: aspectos sociais, gerenciais e de negócio.

Os aspectos sociais envolvem os fatores que podem agir sobre o comportamento e características da comunidade envolvida no ecossistema, ou os fatores que podem ser impactados com este tipo de abordagem. A este grupo pertencem a comunicação, que é diretamente afetada especialmente pelo fato de que os envolvidos no sistema não mais estão localizados no mesmo ambiente físico, devendo considerar outros meios de comunicação que não seja somente face a face; os relacionamentos, que passam a ser mais complexos devido às diversas organizações envolvidas na comunidade que se desenvolvem através de relações de mutualismo e até mesmo competição; a colaboração entre os parceiros considerada a força motriz para o desenvolvimento do ecossistema; e a motivação que pode ser afetada pela distância entre as equipes e conseqüente aumento de responsabilidade e autonomia das mesmas. O grupo gerencial envolve as características inerentes ao surgimento do SECO que dificultam o seu gerenciamento, tais como diferenças culturais, geográficas e de linguagens, já que diferentes equipes podem estar localizadas em qualquer lugar do mundo. Outro aspecto gerencial a ser considerado é a arquitetura, que deve ser decomposta em partes menores, pouco relacionadas entre si, para que o gerenciamento seja facilitado. O último grupo

apresentado é o dos aspectos de negócios, que estão relacionados não com os sistemas envolvidos ou a comunidade, mas sim com as estruturas das organizações envolvidas. Com a mudança da abordagem tradicional para a globalizada é essencial que haja também uma transformação interna de cada empresa envolvida para suportar essa mudança. Novos investimentos devem ser feitos, especialmente no âmbito da comunicação e do gerenciamento. Além disso, deve se considerar os processos internos que devem ser reestruturados já que os tradicionais normalmente não estão adequados a uma estrutura globalizada, envolvendo parceiros externos. Como obstáculo a essas mudanças, está o conhecimento tácito existente em cada organização que pode fazer com que grande parte dos funcionários envolvidos seja resistente às transformações necessárias para adoção da abordagem de ecossistemas.

Conhecer e analisar os aspectos relacionados com o surgimento do SECO e com seu desenvolvimento pode fazer com que as organizações executem planos de contingência e mitigação frente aos obstáculos que poderão surgir, além de possibilitar um maior aproveitamento das possíveis oportunidades para desenvolvimento e prosperidade do ecossistema.

Algumas questões ainda estão em aberto quando se trata desses aspectos. Esta pesquisa abordou o surgimento de um SECO a partir de três perspectivas (social, gerencial e de negócio). Outras perspectivas podem ser analisadas com intuito de conhecer outros impactos que podem ser causados pela mudança de abordagem. Outra questão está relacionada aos planos de gerenciamento de interação e motivação dos *players*, que podem ser melhor trabalhados e formalizados para que as parcerias fechadas trabalhem de maneira mais eficiente e em sincronia, incentivando também a cooperação entre eles.

Bosch (2009) afirma que uma das questões ainda abertas no contexto de ecossistemas de software é a experiência do usuário, que pode ser prejudicada e se tornar menos atraente devido aos sistemas serem desenvolvidos por diferentes partes. Embora *keystones* possam fornecer algum trabalho que minimize este problema, ainda não há uma solução que o resolva completamente.

Outra importante questão ainda em aberto no âmbito de ecossistemas é o fato de que modelos de referência que contêm práticas necessárias à maturidade dos processos,

como o CMMI, se tornam difíceis de serem aplicados, já que equipes externas não podem mais ser submetidas a padrões e modelos de processos e formas de trabalho tradicionais. Definir práticas que levem a maturidade dos processos globalizados é uma tarefa difícil de ser concretizada, mas que se trabalhada poderia tornar os trabalhos dos SECO's mais eficientes e eficazes, podendo, inclusive, definir a melhoria contínua e otimização desses trabalhos, que seriam especialmente importantes na etapa de auto-renovação.

Referências

Andrade, R. **Understanding Strategies During the Software Ecosystems Lifecycle**. Centro de Informática – Universidade Federal de Pernambuco, 2012.

Andrade, R., Alves, C. and Albuquerque, A. **A Model Proposal for Characterizing Software Ecosystems**. Centro de Informática – Universidade Federal de Pernambuco, 2011.

Answers – **Who manufactures the iPhone?** Disponível em: <http://wiki.answers.com/Q/Who_manufactures_the_iPhone?>. Acesso em 04 de junho de 2012.

Barbosa, O. **Understanding Software Ecosystems: Implications for practice and theory**. Centro de Informática – Universidade Federal de Pernambuco, 2012.

Berk, I.M., Jansen, S., Luinenburg, L. **Software Ecosystems: A Software Ecosystem Strategy Assessment Model**. In proceedings of the 4th European Conference on Software Architecture, 2010.

Bosch, J. **From Software Product Lines to Software Ecosystems**. Intuit, 2500 Garcia Avenue, 2009.

Bosch, J., Bosch-Sijtsema, P. From integration to composition: On the Impact of Software Product Lines, Global Development and Ecosystems. **The Journal of Systems and Software** 83, 2010.

Boucharas, V., Jansen, S., Brinkkemper, S. **Formalizing Software Ecosystem Modeling**. Utrecht University, 2009.

Brooks, A. **App Store Downloads Top 2 Billion, 85.000 Apps Available.** Publicado em 28 de setembro de 2009. Disponível em: <<http://www.worldofapple.com/archives/2009/09/28/app-store-downloads-top-2-billion-85000-apps-available/>>. Acesso em 07 de junho de 2012.

Campbell, P., Ahmed, F. **A Three-Dimensional View of Software Ecosystems.** In United Arab Emirates University, Al Ain, UAE, 2010.

Clements, P. **An Introduction to Software Product Lines.** Publicado em 1 de Março de 2005. Disponível em: <<http://www.sei.cmu.edu/library/abstracts/news-at-sei/productlines20053.cfm>>. Acesso em 02 de julho de 2012.

Cooper, P. **Demand for iPhone Developers Up Significantly – Make Your Mark Now!.** Publicado em 30 de outubro de 2008. Disponível em: <<http://mobileorchard.com/iphone-developer-job-demand-up-significantly/>>. Acesso em 04 de junho de 2012.

Farbey, B., Finkelstein, A. **Exploiting software supply chain business architecture: a research agenda.** Department of Computer Science, 1999.

Goeminne, M., Mens, T. **A Framework for Analysing and Visualising Open Source Software Ecosystems.** Software Engineering Lab University of Mons – UMONS, 2010.

Grabianowski, E. **Como funciona a simbiose.** Publicado em 7 de março de 2008. Disponível em: <<http://ciencia.hsw.uol.com.br/simbiose1.htm>>. Acesso em 2 junho 2012.

Iansiti, M., Levien, R. **Strategy as Ecology.** Harvard Business Review, 2004.

Idu, A., Zande, T., Janses, S. **Multi-homing in the Apple Ecosystem: Why and How Developers Target Multiple Apple App Stores.** Department of Information and Computing Science - Utrecht University, 2011.

Ingen, K., Ommen, J., Jansen, S. **Improving Activity in Communities of Practice through Software Release Management.** Utrecht University, 2011.

Jansen, S., Brinkkemper, S., Finkelstein, A. **Business Network Management as a Survival Strategy: A Tale of Two Software Ecosystems.** In proceedings of the 1st Workshop on Software Ecosystems, 2009.

Kilamo, T., Hammouda, I., Mikkonen, T., Aaltonen, T. From proprietary to open source – Growing an open source ecosystem. **The Journal of Systems and Software**, 2010.

Lungu, M., Lanza, M., Gîrba, T., Robbes, R. **The Small Project Observatory: Visualizing software ecosystems.** University of Bern, 2009.

Mendes, S. **Graças ao iPhone, Apple lidera Mercado da telefonia celular.** Publicado em 21 de abril de 2011. Disponível em: <<http://www.portugues.rfi.fr/economia/20110421-gracas-ao-iphone-apple-lidera-mercado-da-telefonia-celular>>. Acesso em 06 de junho de 2012.

Metzger, J. **O que é ecologia de paisagens?** Laboratório de Ecologia de Paisagens e Conservação – LEPaC – Departamento de Ecologia, Instituto de Biociências USP, 2001.

Moore, J.F. Predators and Prey: A New Ecology of Competition. **Harvard Business Review**, 1993.

Power, M.E.; Tilman, D.; Estes, J.A; Menge, B.A; Bond, W.J.; Mills, L.S.; Daily, G.; Castilla, J.C.; Lubchenco, J.; Paine, R.T. **Challenges in the quest for keystones.** Bioscience 609-620.46, 1996.

Scacchi, W. **Free/Open Source Software Development: Recent Research Results and Emerging Opportunities.** Institute for Software Research – University of California, 2007.

Ververs, E., Bommel, R., Jansen, S. **Influences on Developer Participation in the Debian Software Ecosystem.** Utrecht University, 2011.

Weiss, M. **Economics of Collectives.** SCE, Carleton University, 2011.

WORLD RESOURCES INSTITUTE. **People and ecosystems: The fraying web of life.** Report Series. 41p. [http://pubs.wri.org/pubs_pdf.cfm?PubID=3027], 2000.

Yamakami, T. **OSS as a digital ecosystem: A Reference Model for Digital Ecosystem of OSS.** CTO Office, ACCESS, 2010.

Yu, L., Ramaswamy, S., Bush, J. **Software Evolvability: An Ecosystem Point of View.** In Proceedings of the 3rd Workshop on Software Evolvability, 2007.

Yu, L., Ramaswamy, S., Bush, J. **Symbiosis and Software Evolvability.** University of Arkansas at Little Rock, 2008.