

**AVM Faculdade Integrada
Arquitetura e Gestão de Infraestrutura em TI
Adonai Estrela Medrado**

**DEVOPS E AGREGAÇÃO DE VALOR AO NEGÓCIO: SÍNTESE
E ANÁLISE DE PRÁTICAS DAS ORGANIZAÇÕES COM
ATUAÇÃO EM TERRITÓRIO NACIONAL**

**Rio de Janeiro
2015**

**AVM Faculdade Integrada
Arquitetura e Gestão de Infraestrutura em TI
Adonai Estrela Medrado**

**DEVOPS E AGREGAÇÃO DE VALOR AO NEGÓCIO: SÍNTESE
E ANÁLISE DE PRÁTICAS DAS ORGANIZAÇÕES COM
ATUAÇÃO EM TERRITÓRIO NACIONAL**

Trabalho de conclusão de curso apresentado à
AVM Faculdade Integrada no curso de Pós-
Graduação em Arquitetura e Gestão de
Infraestrutura em TI.
Wagner Marcelo Sanchez

**Rio de Janeiro
2015**

Resumo

Este trabalho aborda a temática de DevOps e agregação de valor ao negócio. O objetivo geral é identificar na bibliografia especializada práticas de DevOps que agregaram valor ao negócio das organizações com atuação em território nacional. Utilizou-se a abordagem qualitativa com a opção metodológica de estudo de casos múltiplos. A amostragem foi intencional, levantada a partir da bibliografia já existente. Descreveu-se e analisou-se as experiências com DevOps em quatro empresas: Walmart, AGCO, CI&T e Zendesk. As práticas que agregaram valor ao negócio foram sintetizadas da seguinte forma: 1) desenho colaborativo de soluções; 2) entrega contínua; 3) colaboração para superar desafios; e 4) soluções inovadoras para resolução de problemas. Os resultados indicam que as práticas utilizadas nestas empresas agregaram valor para o negócio nos casos abordados, porém não podem ser generalizadas e adotadas como fórmula descontextualizada e desconectada da cultura organizacional.

Palavras-chave: DevOps, Desenvolvimento, Operações, Agregação de valor

Abstract

This paper addresses the issue of DevOps and added value to the business. The overall objective is to identify in the specialized bibliography DevOps practices that added value to the business of organizations with operations in Brazil. We used a qualitative approach with the methodological option of multiple case studies. Sampling was intentional, gathered from the existing literature. It was described and analyzed the experiences with DevOps in four companies: Walmart, AGCO, CI & T and Zendesk. The practices that added value to the business were summarized as following: 1) collaborative design solutions; 2) continuous delivery; 3) collaboration to overcome challenges; and 4) innovative solutions for problem solving. The results indicate that the practices used in these organizations added value to the business in the cases covered, but cannot be generalized and adopted as decontextualized and unplugged formulas from the organizational culture.

Keywords: DevOps, Development, Operations, Added value.

Sumário

INTRODUÇÃO	5
CAPÍTULO 1 - COLABORAÇÃO COMO CAMINHO PARA ENTREGAR VALOR AO NEGÓCIO	9
1.1 Nascimento do termo.....	10
1.2 Conflito entre projeto e operação.....	11
1.3 Automação para reduzir risco e aumentar a agilidade.....	12
1.4 Uma nova cultura para desenvolver soluções	14
1.5 Princípios fundamentais.....	17
1.6 Níveis de maturidade	18
1.7 Retorno do investimento.....	21
CAPÍTULO 2 - DEVOPS EM CASOS	23
2.1 Walmart: DevOps integrando equipes e entregando soluções	23
2.2 AGCO: da fábrica ao DevOps.....	25
2.3 CI&T: DevOps na nuvem.....	28
2.4 Zendesk: DevOps com equipes geodistribuídas.....	30
CAPÍTULO 3 - VALOR PARA O NEGÓCIO, VISÕES E SOLUÇÕES	34
3.1 Concepção de DevOps.....	36
3.2 Estímulo inicial.....	41
3.3 Principais ferramentas utilizadas	44
3.4 Resultados alcançados.....	48
CONCLUSÃO.....	51
REFERÊNCIAS.....	54

INTRODUÇÃO

Na área de Tecnologia da Informação (TI), designa-se DevOps o movimento de aproximação entre as equipes de desenvolvimento e operações (LOUKIDES, 2012). O termo é um amálgama das palavras desenvolvimento e operações, abrangendo uma prática pautada na colaboração e na cooperação que tem atualmente atraído grande interesse.

Quase todas as grandes empresas de TI possuem, em seus sites oficiais, textos informativos que associam seus produtos às práticas relacionadas a este conceito. Dentre alguns exemplos pode-se citar a Microsoft com o Team Foundation Server e a IBM com o IBM UrbanCode Deploy.

Observa-se uma produção literária que também aborda produtos e ferramentas voltadas para DevOps. Tal produção tanto é patrocinada pelos fabricantes – como Guthrie et al (2014) que aborda as melhores práticas para DevOps considerando o ambiente Microsoft Azure – quanto é realizada por analistas independentes – como Thurman (2013) que faz uma breve análise de cinco ferramentas pouco reverberadas no meio.

Entretanto, como observa Loukides (2012) e Walls (2014), DevOps é mais do que um conjunto de ferramentas facilitadoras. A implantação de suas práticas exige uma mudança na cultura organizacional e se justifica mediante uma agregação de valor ao negócio.

Apesar disto, a maioria do material encontrado na literatura online e offline discute apenas conceito e ferramenta. Há exceções, como o relatório elaborado pela Puppetlabs (2014) o qual, no contexto de DevOps, alega que uma organização com uma TI performática teria duas vezes mais chances de aumentar seus lucros, sua fatia no mercado e sua meta de produtividade. Este relatório pode, no entanto, ser considerado tendencioso por três aspectos principais: 1) foi patrocinado por uma empresa que comercializa produtos relacionados à DevOps, portanto diretamente interessada nos seus resultados; 2) foi elaborado a partir de uma pesquisa entre

profissionais da área de TI, normalmente preocupados em valorizar a importância de seus departamentos; e 3) não foram solicitadas evidências das respostas fornecidas, dando margem para a subjetividade, supervalorização e má fé.

Desta forma, constata-se uma carência de material neutro, capaz de fornecer insumos para que gestores decidam investir recursos organizacionais na implantação de práticas de DevOps. Este trabalho colabora para preencher esta lacuna literária, buscando abordar a temática de forma a contribuir com os subsídios necessários ao processo decisório.

Tema

DevOps e agregação de valor ao negócio: síntese e análise de práticas das organizações com atuação em território nacional.

Problema

Quais práticas de DevOps relatadas na bibliografia especializada agregaram valor ao negócio das organizações com atuação em território nacional?

Justificativa

Em um mundo empresarial cada vez mais competitivo, em que as organizações estão empenhadas num processo de melhoria contínua da qualidade dos seus produtos e serviços, torna-se essencial o uso racional dos recursos disponíveis. As expectativas dos clientes definem tanto o valor atribuído ao atendimento quanto onde, quando e como ocorrerão os investimentos empresariais (SILVA; RIBEIRO, 2002). Neste contexto, agregar valor ao negócio significa montar uma estratégia de sobrevivência e crescimento empresarial capaz de atender e se adequar às expectativas do cliente.

A adoção de uma estratégia organizacional precisa ir além do modismo ou da antropofagia, ultrapassando o simples aplicar de conceitos estrangeiros, sem ressignificação ou recontextualização (WOOD JR.; CALDAS, 1998). Em meio aos métodos ágeis de desenvolvimento de software, DevOps insere-se como um conjunto

de conceitos que rapidamente estão se espalhando no meio técnico e que pretendem colaborar para entrega contínua de soluções de software.

Integrar novos conceitos no meio empresarial significa investir recursos em treinamento, em capacitação e em uma mudança da cultura organizacional. Desta forma, este trabalho se justifica na medida em que se propõe a levantar na literatura especializadas quais práticas de DevOps agregam valor ao negócio, contribuindo para uma decisão consciente de como priorizar os investimentos.

Objetivo geral

Identificar na bibliografia especializada práticas de DevOps que agregaram valor ao negócio das organizações com atuação em território nacional.

Objetivos específicos

1. Conceituar DevOps contextualizando-o historicamente;
2. Relacionar o uso de DevOps à cultura organizacional;
3. Sintetizar e analisar práticas de DevOps relatadas pela comunidade especializada nas organizações com atuação em território nacional;
4. Correlacionar o conceito de DevOps inserido na cultura organizacional aos resultados alcançados pelas suas práticas, identificando aquelas que agregaram valor ao negócio.

Metodologia

Para atingir os objetivos deste trabalho, optou-se pela abordagem qualitativa utilizando a opção metodológica de estudo de casos múltiplos pela qual se busca a comparação e compreensão do fenômeno com base em vários casos. A abordagem qualitativa coleta os dados em um contexto natural, sensível às pessoas e aos lugares de estudo. Nela abordam-se os significados que indivíduos ou grupos atribuem a um problema social ou humano. No relatório final de uma pesquisa qualitativa, estão inclusas as reflexões do pesquisador, a descrição do problema e uma avaliação das contribuições para a literatura da área. (CRESWELL, 2014)

Um estudo de caso pode ser definido como uma investigação empírica que trata um fenômeno contemporâneo dentro de seu contexto, sendo adequado, principalmente, quando fenômeno e contexto não estão claramente delimitados (YIN, 2001). Um dos motivos para escolha deste método é que ele permite lidar com condições contextuais (como a cultura organizacional), extremamente relevantes para o fenômeno DevOps.

A amostragem será intencional, levantada a partir da bibliografia já existente, de forma que o conjunto se preste a estudar diferentes práticas de DevOps. A preferência será por casos que possam ser corroborados com mais de uma fonte de informação e que possuam aspecto restrito, com descrição de um processo organizacional específico.

Utilizar-se-á a análise incorporada. Assim, conforme descrito em Creswell (2014) e Yin (2001), cada caso será primeiramente detalhado e em seguida serão focadas questões-chave para os objetivos do trabalho (análise dentro do caso). Posteriormente realizar-se-á a análise cruzada, com interpretação dos significados dos casos apresentados. No procedimento analítico, será comparada a prática no ambiente organizacional àquilo que é esperado em DevOps, bem como serão identificados os objetivos alcançados no que diz respeito à agregação de valor ao negócio.

CAPÍTULO 1 - COLABORAÇÃO COMO CAMINHO PARA ENTREGAR VALOR AO NEGÓCIO

No início da era do computador, não havia distinção entre as equipes que desenvolviam soluções e as equipes que as mantinham em operação, pois o mesmo indivíduo desempenhava ambos os papéis. Somente a partir dos anos 60, com o uso dos mainframes, é possível observar uma separação entre as funções de desenvolvedor (responsável pela escrita dos códigos fontes) e operador (responsável pela montagem das fitas, distribuir as impressões resultantes de uma execução etc.). Com o advento dos microcomputadores, modificou-se a forma como cada equipe desempenhava seu papel. Os desenvolvedores passaram a ter um domínio das suas próprias máquinas, mas tinham que compartilhar os recursos dos servidores. A equipe de operações, agora chamados de administradores de sistema ou administradores de rede, ficou responsável por manter estes recursos disponíveis para serem utilizados quando fossem necessários. (LOUKIDES, 2012)

Com o crescente uso das tecnologias de informação, foi necessário uma resignificação de papéis. Os desenvolvedores precisavam pensar em formas de desenvolver suas soluções com maior agilidade e os administradores de sistema já não podiam manter centenas de servidores da mesma forma como faziam com meia dúzia. Era preciso automatizar, garantir uma operação confiável através de padrões e práticas escaláveis, reproduzíveis e programáveis. A infraestrutura de trabalho do operador deixou de ser apenas hardware; um recurso (máquina, porta, insumos de rede etc.) pode ser virtual, ou seja, a infraestrutura está se transformando em código de programação. Um administrador de sistemas neste novo paradigma precisa interagir com este mundo, modificando a sua forma de trabalho: ao invés de trocar peças, é preciso escrever códigos capazes de identificar problemas e corrigi-los sem a interrupção do serviço. (LOUKIDES, 2012)

Não há uma extinção dos operadores, mas uma aproximação de suas atividades com as atividades dos desenvolvedores e vice-versa, visto que, no desenvolvimento de soluções de software, é necessário o conhecimento dos recursos tecnológicos,

expertise para otimizá-los e mantê-los tolerante a falhas. Portanto, no contexto contemporâneo, as equipes de desenvolvimento e operações precisam colaborar e cooperar para criar aplicações de software que não só atendam aos seus requisitos funcionais, mas que sejam administráveis e utilizem os recursos de infraestrutura da melhor forma possível. O movimento é de aproximação e demanda que especialistas em operações integrem-se à equipe de desenvolvimento naquilo que passou a ser conhecido como DevOps. (LOUKIDES, 2012)

1.1 Nascimento do termo

Empolgado com o seminário “*10+ Deploys Per Day: Dev and Ops Cooperation at Flickr*”¹ de John Allspaw no Velocity 2009, Patrick Debois organizou uma conferência de dois dias em Gante (Bélgica) no final de 2009 a qual deu o nome de *DevOps Days*. Muitos dos menos de cinquenta participantes organizaram em seguida *DevOps Days* nos seus próprios países e começaram a difundir o tema utilizando blogs. Em 2010 foi organizado o *DevOps Days* em Mountain View nos EUA. (WILLS, 2012) Trabalhos anteriores de O’Reilly (2006) e Robbins (2007) merecem destaque, pois já sinalizavam para a relevância da equipe de operações.

Outros movimentos importantes para DevOps foram o “*Infrastructure as code*” – Mark Burgess and Luke Kanies –, “*Agile infrastructure*” – Andrew Shafer –, “*Agile system administration*” – Patrick DeBois –, “*Lean Startup*” – Eric Ries –, “*Continuous integration and release*” – Jez Humble – além da expansão dos serviços de nuvem e plataforma como serviço (*Plataform as a Service* – PaaS). (KIM, 2012)

No contexto geral de TI, o movimento de DevOps é complementar ao Desenvolvimento Ágil, inserindo-se no processo de integração e entrega contínua, assegurando que o código está pronto para produção e provendo valor para o cliente além de permitir um melhor fluxo de trabalho e integração entre o desenvolvimento e operações. (KIM, 2012)

¹ A sinopse e a apresentação deste seminário estão disponíveis no seguinte endereço Web: <<http://velocityconf.com/velocity2009/public/schedule/detail/7641>>. Acesso em 04 jul. 2015.

1.2 Conflito entre projeto e operação

Há desafios a esta integração que estão no cerne das atividades e objetivos das equipes envolvidas. Enquanto o desenvolvimento trabalha no sentido da evolução dos sistemas através de alterações no software, a equipe de operações tenta evitar mudanças, pois elas põem em risco a estabilidade do sistema. A maneira mais comum de lidar com esta relação por natureza conflitante é criar processos de forma a atender às necessidades de ambas as equipes, porém, com o decorrer do tempo, a burocracia envolvida tende a ser cada vez maior, o que tem como consequência a redução da frequência com que as modificações de software são implantadas em ambiente produtivo. (SATO, 2014)

À primeira vista, a redução da frequência de implantações parece positiva do ponto de vista da equipe de operações, entretanto o que se percebe neste caso é o aumento do risco de instabilidade, pois, com a redução da frequência, a quantidade de modificações implantadas de uma só vez é maior. Do ponto de vista de entrega de valor ao negócio, é importante destacar que só há potencial geração de valor quando o software é implantado em ambiente produtivo. Quanto mais tempo uma modificação demora a ser efetuada em produção, maior será o tempo para obter o retorno do investimento. (SATO, 2014)

Atrasar uma implantação em produção significa também atraso na capacidade de competir e adaptar-se às mudanças de mercado, o que justifica o crescimento de uma estratégia que ficou conhecida como Entrega Contínua (*Continuous Delivery*) que visa uma entrega mais rápida das alterações e evoluções do software ao negócio. Entregas mais rápidas têm como consequência menor número de alterações por implantação e menor risco à estabilidade do sistema, porém demandam simplificação dos canais de comunicação entre as equipes de desenvolvimento e operações. Isto não significa fim do processo de interação entre estas duas equipes, mas um investimento na automação deste processo. (SATO, 2014)

O movimento DevOps tem por objetivo criar uma cultura da colaboração onde exista a percepção de agilidade no fluxo de entrega das soluções demandadas pelo negócio e

que, ao mesmo tempo, aumente a estabilidade e confiança no ambiente produtivo. Nesta direção, a área de Tecnologia de Informação da organização pode deixar de ser percebida como um gargalo da companhia, que demora a disponibilizar as soluções e alterações demandadas pelo mercado, passando a potencializar competitivamente o negócio. (SATO, 2014)

1.3 Automação para reduzir risco e aumentar a agilidade

Para potencializar competitivamente o negócio, não basta entregar soluções rapidamente, é preciso atender requisitos de qualidade, o que geralmente envolve uma bateria de testes de maneira que o processo de implantação em ambiente produtivo seja feito com confiabilidade e com o menor risco possível.

Um dos padrões da Entrega Contínua para suplantar este desafio é a trilha de implantação (*deployment pipeline*). A trilha de implantação é a automação do processo de construção, implantação, teste e liberação. Toda alteração realizada em uma configuração, código fonte, ambiente ou dado ativa uma instância da trilha. As organizações podem implementar esta trilha de formas distintas, mas usualmente um dos primeiros passos é a criação dos binários e do instalador, depois são executadas uma série de testes para provar que estes arquivos podem ser liberados para a implantação. A importância dos testes está na aferição de certo grau de confiabilidade nas modificações realizadas. (FARLEY; HUMBLE, 2010)

Os objetivos da trilha de implantação são três: 1) dar transparência ao processo, pois os resultados de cada etapa da trilha estão disponíveis para todos os envolvidos; 2) aumentar o feedback de forma que problemas sejam identificados e resolvidos o mais cedo possível; 3) possibilitar à equipe implantar qualquer versão do software em qualquer ambiente por meio de um processo totalmente automatizado. (FARLEY; HUMBLE, 2010)

Sem automação, há vários fatores que podem influenciar negativamente o processo de implantação, uma vez que as atividades manuais, quando não são executadas com perfeição, podem gerar erros difíceis de serem identificados e tomar um tempo desnecessário das equipes. Por exemplo, um processo de implantação manual

geralmente exige a produção de uma extensa e detalhada documentação que descreve os passos a serem tomados, mas ainda assim são frequentes as ligações para o time de desenvolvimento para tentar identificar o que ocorreu de errado na implantação. Em um dos piores cenários, a implantação não funciona ou roda de forma não prevista devido a erros na documentação e/ou na execução do procedimento. (FARLEY; HUMBLE, 2010)

Por outro lado, as implantações automatizadas são repetíveis e encorajam a colaboração, pois tudo está em um conjunto de scripts. Elas também não dependem da expertise de um grupo de pessoas. As únicas tarefas a serem executadas manualmente são 1) escolher a versão a ser implantada e o ambiente de destino; 2) enviar o comando para implantação – que pode ser o simples clique de um botão. (FARLEY; HUMBLE, 2010)

É recomendável que o processo de implantação automático seja utilizado por todos aqueles que necessitam implantar o software, independente do ambiente de destino. Nesta mesma linha, é também recomendável que a implantação automática seja a única forma possível de se realizar a implantação e que o script utilizado seja o mesmo para todos os ambientes (desenvolvimento, teste, homologação e produção), desta forma, quando for necessário executar a implantação em produção, o script já terá sido testado diversas vezes, assim, se algo der errado, será bastante provável que o problema esteja nas configurações específicas para o ambiente produtivo. (FARLEY; HUMBLE, 2010)

A adoção da trilha de implantação otimiza e dinamiza o processo de implantação. A previsibilidade e a possibilidade de entregar ao negócio com agilidade e confiabilidade versões do software são ganhos importantes desta estratégia, porém um pré-requisito para a implantação deste processo é uma cultura de colaboração entre as equipes de infraestrutura e operações. Por exemplo, os scripts de automação precisarão executar com um nível de privilégio que usualmente apenas a equipe de operações possui em ambiente produtivo. É necessária a construção de uma prática colaborativa onde exista confiança e dialogo entre as equipes de forma a alavancar as soluções requeridas pelo negócio.

1.4 Uma nova cultura para desenvolver soluções

Esta prática colaborativa imbrincada no termo DevOps exige repensar a cultura organizacional. Indivíduos, equipes, organização e clientes precisam construir um novo alinhamento com relação à maneira de desenvolver soluções. Não há mais lugar para o antigo “jogo de empurra” em que a culpa de uma solução não funcionar era constantemente deslocada entre a equipe de desenvolvimento e a equipe de operações. Agora ambas estão juntas e são responsáveis por possibilitar o negócio e criar valor para o cliente. (WALLS, 2014)

O conceito de cultura organizacional originou-se na metade do século XX e constitui-se na ideia de que um grupo de pessoas que trabalha junto num ambiente organizacional pode compartilhar um conjunto de valores e comportamentos que o distingue. A maneira como a cultura organizacional é percebida pelos sujeitos, afeta como cada um irá reagir às tentativas de mudança. Apesar de não existir uma receita aplicável a todos os casos, a incorporação de práticas de DevOps terá maior probabilidade de sucesso num ambiente cultural favorável para colaboração e cooperação, onde se valorize a comunicação aberta, o alinhamento de responsabilidades, o respeito e a confiança. (WALLS, 2014)

O conflito entre a equipe de desenvolvimento e operações leva a um cenário onde a equipe de TI atua como dois silos nos quais a colaboração fica em segundo plano. O movimento de DevOps estreita esta interação com o incentivo do compartilhamento de processos e ferramentas, aproximando estas duas equipes, utilizando-se de uma variedade de práticas para que a TI entregue valor e mantenha-se alinhada ao negócio. (HÜTTLERMANN, 2012)

As práticas em DevOps dinamizam o processo de entrega de software, mas perpassam por fatores culturais, pois as soluções de TI, assim como os processos organizacionais, é feito pelo homem e para o homem, não podendo ser desvinculado da cultura (HÜTTLERMANN, 2012). Designar uma ferramenta ou abordagem como alinhada ao movimento de DevOps não significa muito, caso sua adoção não esteja alinhada e refletida na cultura organizacional usualmente influenciada pelas métricas vigentes.

A equipe de desenvolvimento geralmente é medida pela velocidade que desenvolve as soluções solicitadas pelo negócio, enquanto a equipe de operações tem o seu sucesso atrelado à disponibilidade e ao tempo de resposta da aplicação. Há, portanto, um conflito entre a cultura da mudança e a do medo da mudança. O desenvolvimento e o negócio desejam as mudanças implantadas no ambiente produtivo, por outro lado a equipe de operações e também o negócio desejam manter a disponibilidade e o tempo de resposta da aplicação dentro das metas estabelecidas. Trabalhando individualmente, a regra será o conflito entre estas equipes. (HÜTTLERMANN, 2012)

O fator de sucesso para DevOps é uma cultura do compartilhamento e do companheirismo, com a colaboração entre desenvolvimento e operações. É necessário aceitar a natureza conflituosa das duas equipes, mas fazer com que todos os interessados no processo de implantação estejam alinhados, trabalhando juntos e compartilhando um mesmo objetivo: entregar valor para o negócio. (HÜTTLERMANN, 2012)

Esta cultura do compartilhamento não significa que o movimento de DevOps encoraja que os desenvolvedores trabalhem diretamente no ambiente de produção nem se trata de criar um departamento de DevOps, tampouco implica necessariamente na adoção de novas ferramentas. DevOps é um movimento que lida com o conflito natural entre desenvolvimento e operações, mas que não trabalha com um conjunto de receitas prontas. Ao invés disto, pessoas, processos e ferramentas formam uma estrutura colaborativa que dinamizam o processo de entrega de software. (HÜTTLERMANN, 2012)

As pessoas são o elo mais importante e as organizações que estão interessadas em DevOps se beneficiam contratando profissionais que resistem à estruturação de silos – geralmente indivíduos com características polímatas e generalistas. Após contratação, o gerente tem um papel fundamental em manter o engajamento profissional, o qual está associado ao comprometimento com as realizações e é fortalecido pelo acompanhamento do desempenho e com exercício de atividades nas quais o empregado possui performance e interesse ótimo. (ZWIEBACK, 2014)

DevOps é principalmente um desafio de gestão e das pessoas. Não são ferramentas que corrigem os problemas da relação entre desenvolvimento e operações. Motivações e processos conflitivos entre estas duas equipes são usualmente encontrados até nas pequenas organizações. O cenário se complica mais quando desenvolvedores e operadores usam ferramentas diferentes, reportam-se a lideranças corporativas distintas e praticam políticas corporativas conflitivas. A desconexão entre as equipes pode chegar ao extremo da equipe de desenvolvimento desconhecer informações básicas do ambiente produtivo – como o sistema operacional. (PHILLIPS, 2013)

DevOps trata-se de fazer as pessoas trabalharem melhor compartilhando conhecimento, objetivo e responsabilidades. Para isto, é preciso refletir sobre uma série de questões, dentre as quais podemos citar: 1) os membros da equipe de desenvolvimento conhecem os membros da equipe de operações?; 2) a equipe de operações é envolvida nos novos projetos de desenvolvimento?; 3) a equipe de operações sabe qual desenvolvedor é responsável por uma aplicação?; 4) há eventos regulares no nível organizacional que integrem as duas equipes? (PHILLIPS, 2013)

Em termos de processo, há outras questões que podem ser feitas e que indicam práticas organizacionais que favorecem ou não DevOps: 1) é possível alocar durante a fase de desenvolvimento uma equipe de operações para colaboração e discussão?; 2) a equipe de desenvolvimento pode reunir-se facilmente com a equipe de operações?; 3) durante a fase de planejamento de uma nova infraestrutura, a equipe de operações pode alocar tempo da equipe de desenvolvimento para discutir as implicações no desenvolvimento?; 4) a equipe de operações pode facilmente conseguir contato com a equipe de desenvolvimento para discussões?; 5) existe compartilhamento de objetivos, métricas ou outros incentivos em que a equipe de desenvolvimento e operações precise colaborar para atingi-los?; 6) desenvolvedores e operadores veem-se trabalhando um com o outro ou como provedores de serviço? (PHILLIPS, 2013)

DevOps não é um problema de ferramentas, mas sim uma questão de encontrar uma maneira de quebrar as barreiras técnicas e culturais que dividem desenvolvedores e operadores. Assim, as preocupações com as ferramentas vão muito mais nesta direção e podem ser assim elencadas: 1) existe uma ferramenta de colaboração na qual o

gestor possa ver de forma compartilhadas, regular e atualizada as atividades das equipes de desenvolvimento e operações?; 2) existe uma ferramenta que provê uma visão compartilhada em que todos possam ver o status e o progresso de uma nova versão do produto que será implantada em produção?; 3) existe uma ferramenta que permite não especialistas em determinada área executarem de forma automática as tarefas que necessitam?; a equipe de operações consegue disparar a compilação de uma versão específica?; um profissional de testes consegue disparar uma implantação? (PHILLIPS, 2013)

Uma vez que o gestor tome a decisão de implantar práticas de DevOps, o desafio é identificar onde investir esforços para obter os ganhos pretendidos. Por exemplo, caso se verifique que a equipe de desenvolvimento está lenta e consumindo muitos recursos abrindo tickets com solicitações para a equipe de operação, poder-se-ia investir em automatização. Por outro lado, caso a equipe de desenvolvimento careça de domínio tecnológico em uma determinada infraestrutura, poder-se-ia integrar um especialista desta área à equipe. Ou seja, os gestores que precisam identificar práticas de DevOps com potencial para agregar valor ao seu negócio de acordo com o contexto em que está inserido.

Cabe destacar que, por ordem de importância, os elementos para o sucesso DevOps são pessoas, processos e ferramentas (HÜTTLERMANN, 2012). Estes três itens fazem parte e influenciam a cultura organizacional e formam uma tríade que precisa de uma gestão adequada para que guie um movimento em direção aos fundamentos de DevOps.

1.5 Princípios fundamentais

Todos os padrões (*patterns*) de DevOps podem ser derivados daquilo que ficou conhecido como “Os três caminhos” (*The Three Ways*) que são utilizados para descrever os valores e a filosofia que guia as práticas e processos de DevOps (KIM; BEHR; SPAFFORD, 2013).

O primeiro caminho (pensamento sistêmico – *systems thinking*) trata do fluxo do trabalho que vai da equipe de desenvolvimento (que implementa as mudanças

demandadas pelo negócio) à equipe de operações (que garante a entrega das mudanças ao cliente). Para o movimento de DevOps, a ênfase é na performance do sistema organizacional como um todo e não na performance de um silo dentro da empresa. Em outras palavras: o foco é no valor para o negócio que é criado pela TI. (KIM, 2012)

O segundo caminho (amplificação do ciclo que feedback – *amplify feedback loops*) é sobre criar um ciclo de feedback. Desenvolvedores devem fornecer feedback para a equipe de operação e vice-versa. Trata-se de entender e responder a todos os clientes (internos e externos), encurtando e amplificando o ciclo de feedback. (KIM, 2012) No cerne deste princípio está a ideia que desenvolvedores podem agregar valor à equipe de operações assim como a equipe de operações pode contribuir para o trabalho da equipe de desenvolvimento.

O terceiro caminho (cultura da experimentação e aprendizado contínuo – *culture of continual experimentation and learning*) trata de incentivar a experimentação e a compreensão que repetição e prática são pré-requisitos para a excelência. Isto requer assumir riscos e aprender com os sucessos e as falhas. (KIM, 2012) Este caminho traz um interessante desafio para gestão que é estimular a inovação e a aprendizagem com vista a obter ganhos para o negócio, mas também mantendo sua equipe motivada e encorajando-a na direção do novo e da mudança – o que pode alavancar a melhoria contínua dos processos, colaborando para elevar a maturidade das soluções entregues ao negócio e do próprio ambiente de DevOps.

1.6 Níveis de maturidade

A Hewlett-Packard (HP) – empresa que comercializa ferramentas que podem ser utilizadas para auxiliar práticas de DevOps – desenvolveu um modelo que descreve cinco níveis de maturidade das organizações que buscam seguir o movimento de DevOps.

Os cinco níveis definidos estão sintetizados no Quadro 1 (página 21) e são os seguintes: inicial, gerenciado, definido, mensurável e otimizado. Em cada nível são analisadas as dimensões de colaboração, automação e processo. Para seguir para o

próximo nível é essencial atingir as três dimensões do nível atual. É importante observar que numa mesma organização é possível que diferentes equipes/atividades estejam em níveis diferentes de maturidade. (HP, 2013)

No nível inicial, os processos geralmente são *ad-hoc* e caóticos, seus resultados não são definidos e é comum exceder o orçamento e o prazo acordado. Há tendência de abandonar os processos em tempos de crise e é impossível repetir o sucesso. Não existe automação, fazendo com que os processos não possam ser repetíveis, tenham baixo nível de controle e sejam lentos. A comunicação e a coordenação são pobres e *ad-hoc*, os papéis e responsabilidades não são bem definidas. As informações são fornecidas utilizando os canais de comunicação formal e as decisões são tomadas de forma independente pelos *stakeholders* e depois comunicada às equipes. (HP, 2013)

No segundo nível, os processos são geridos, mas não padronizados entre projetos ou mesmo durante o ciclo de vida de um mesmo projeto. Grupos diferentes podem utilizar processos diferentes. A automação é parcial e não possui uma infraestrutura centralizada. A comunicação e coordenação é gerida, há reuniões de alinhamento, uma liberação é comunicada e coordenada entre as equipes de desenvolvimento e operação as quais compartilham informações e, em alguns casos, recursos. O papel de *stakeholder* é bem definido, a comunicação entre as equipes é frequente e algumas tomadas de decisão são compartilhadas, mas a maioria ainda é tomada isoladamente para depois ser comunicada e coordenada com as equipes. (HP, 2013)

No terceiro nível, os processos estão bem caracterizados e padronizados entre projetos. Entretanto, cada projeto estabelece os seus processos específicos, modificando o processo padrão para atender as suas necessidades e requisitos. Existe uma infraestrutura central de automação que suporta toda a organização. A colaboração é estabelecida entre as equipes, tornando-se uma parte essencial dos processos padronizados. Há um conjunto de ferramentas que possibilita a troca de ideias, transparência e rápido feedback. Todos os membros das equipes pertencem a um único sistema com responsabilidades compartilhadas, comunicação frequente e confiança mútua. (HP, 2013)

No quarto nível, a qualidade e o desempenho do processo são medidas para obter transparência e previsibilidade. O processo tem seu desempenho controlado utilizando métodos quantitativos e a previsibilidade é baseada na análise estatística de dados. Processos automatizados fim a fim são medidos e controlados promovendo transparência e detalhamento (status, custo, tempo para completar, stakeholders) para cada atividade. As métricas dos processos automatizados são medidas em relação aos objetivos do negócio. Processos baseados em colaboração são medidos para identificar ineficiências e gargalos com base nas informações das pessoas envolvidas nas atividades (incluindo seus níveis de expertise e contribuição, relevância da informação fornecida ou nível de eficiência durante a execução). (HP, 2013)

No quinto nível, há uma avaliação contínua de todo o processo buscando atender aos objetivos do negócio com o menor risco e custo. Os processos são continuamente melhorados através de análise de métricas, autoaprendizado e autorremediação. Autosserviço é disponibilizado para diferentes *stakeholders*. A colaboração é otimizada para efetividade e compartilhamento continuado do conhecimento com empoderamento dos indivíduos. (HP, 2013)

Este modelo de maturidade é útil por considerar dimensões importantes de DevOps e por poder ser empregado num processo de autoavaliação organizacional. Gestores podem utilizá-lo como referência para melhorar e investir em determinadas práticas, sendo possível inclusive traçar um caminho para que este investimento seja feito de forma consciente. Porém, não se deve esquecer nem dos princípios fundamentais, nem da ordem de importância dos elementos essenciais para o sucesso em DevOps, ou seja, primeiro pessoas, depois processos para só então ferramentas.

	Processo	Automação	Colaboração
Inicial	Imprevisível, sem controle e reativo.	Sem automação	Fraca, comunicação e coordenação <i>ad-hoc</i> .

Gerenciado	Gerenciado, mas não padronizado.	Automação em silos, sem infraestrutura central de automação.	Comunicação gerenciada, algumas decisões compartilhadas.
Definido	Padronizado em toda a organização.	Processo de automação central acompanhando o ciclo de vida da aplicação.	Colaboração, decisões e responsabilidades compartilhadas.
Mensurável	Qualidade e performance transparente e previsível.	Coleta e analisa métricas da automação do processo comparando com os objetivos de negócio.	Processos baseados em colaboração são mensuráveis para identificar ineficiências e gargalos.
Otimizado	Otimização de risco e custo do processo.	Autosserviço automatizado, autoaprendizado utilizando estatísticas e autorremediação.	Compartilhamento efetivo de conhecimento com empoderamento individual.

Quadro 1: Níveis de maturidade em DevOps - adaptado de HP (2013)

1.7 Retorno do investimento

Os resultados indicam que esta forma de estruturação das equipes traz ganhos às organizações. A prática de DevOps pela empresa de vídeos sob demanda Netflix é citada com frequência na literatura especializada. Loukides (2012), por exemplo, faz referência à Netflix's ChaosMonkey, uma ferramenta desenvolvida como parte do

esforço colaborativo entre desenvolvedores e operadores para testar e garantir que a aplicação consiga suportar falhas sem degradação da qualidade do serviço. Cockcroft (2012), empregado da Netflix, relata os benefícios de terem um time de especialistas na tecnologia de nuvem integrados na equipe de desenvolvimento. Para ele, este time de especialista ajudou-os a ter maior domínio na tecnologia para a qual a empresa estava migrando sua plataforma web e de streaming.

No caso Netflix, os relatos indicam que DevOps agregou valor ao negócio, pois possibilitou uma melhora na qualidade de pelo menos um atributo valorizado pelos seus clientes (a saber: serviço de vídeo sob demanda com pouca ou nenhuma instabilidade). Aumentar a satisfação do cliente exige, dentre outros aspectos, uma organização interna dos processos e uma redução da discrepância entre expectativas e impressões (SILVA; RIBEIRO, 2002). Assim, embora existam inúmeras ferramentas de software que podem estar inclusas na prática de DevOps (ex.: Chef, Puppet, Git, Jenkins), é uma mudança na relação e no processo entre as equipes de desenvolvimento e operações que beneficia o negócio.

No próximo capítulo serão sintetizadas práticas de DevOps relatadas pela comunidade especializada nas organizações com atuação em território nacional de forma a compreender os resultados alcançados com os investimentos realizados, especificamente no que diz respeito ao valor agregado ao negócio.

CAPÍTULO 2 - DEVOPS EM CASOS

Durante a coleta dos casos apresentados neste capítulo, percebeu-se que o termo DevOps exhibe significados múltiplos dentro da comunidade especializada. Nem todos os casos descritos compartilham uma mesma conceituação do termo, porém tomou-se a opção metodológica de assumir a autodeclaração como suficiente para enquadramento do caso, uma vez que o nível de maturidade da organização pode influenciar sua percepção sobre o conceito.

Nos casos, perceber-se-á que o estímulo inicial para adotar práticas de DevOps também variou bastante, entretanto sempre houve uma relação com alguma motivação do negócio.

No decorrer da descrição, citar-se-á uma série de ferramentas de automação utilizadas pelas equipes envolvidas. Optou-se por não entrar em detalhes funcionais de cada ferramenta, mas, quando possível, explicitar o seu uso no contexto específico. Na primeira vez em que o nome de uma ferramenta aparecer no texto, uma nota de rodapé indicará o endereço de seu site oficial.

Em todos os casos descritos, os resultados alcançados foram positivos em sua maioria. Durante a fase de coleta, não foi localizado caso em que houve domínio de resultados negativos.

As empresas relacionadas possuem equipes no Brasil, porém são multinacionais com atuação global. Uma descrição detalhada do negócio da empresa e de sua situação no mercado foge ao escopo deste trabalho, mas o link para página oficial da organização ficará em nota de rodapé para maiores detalhes.

2.1 Walmart: DevOps integrando equipes e entregando soluções

Lorieri (2015) apresentou um caso de uso de DevOps envolvendo a empresa Walmart² e a Vtex³. Para ele, DevOps significa ter comprometimento e objetivo comum. Estas duas características levariam à colaboração, à transparência e a uma gestão de riscos

² Site oficial no Brasil: <http://www.walmart.com.br/>

³ Site oficial: <http://www.vtex.com/>

mais inteligente. O medo de errar e de novos problemas seria um dos motivos que impede a implantação de DevOps.

Após compra do *marketplace* da Vtex, a equipe de TI da Walmart foi solicitada a implantar este produto, que normalmente era oferecido como uma plataforma na nuvem (*Plataform as a Service – PaaS*), na infraestrutura do próprio Walmart. O desafio era trabalhar com cerca de 20 módulos desenvolvidos na plataforma Microsoft .NET configurados com alta disponibilidade.

A Vtex continuaria a suportar o sistema na infraestrutura da Walmart e para tanto precisava que fossem mantidas as mesmas características do seu sistema oferecido no modelo PaaS. Na percepção de Lorieri (2015), na nuvem, a Vtex mantinha um único sistema atendendo a diversas lojas de pequeno e médio porte, trabalhava com times distribuídos autônomos que tinham agilidade para lidar com riscos altos e realizar centenas de implantações diárias. Já a cultura do Walmart era de uma grande loja, que trabalhava com virtualização, com times especializados (desenvolvimento, infraestrutura, redes etc.) que lidava com riscos baixos para evitar desvalorização da marca e realizava dúzias de implantações diárias.

Esta diferença cultural deixava o sistema comprado pela Walmart desatualizado e sem sincronia com o sistema ofertado pela Vtex na nuvem, além de dificultar os processos de homologação e implantação. Com o objetivo de encontrar uma solução, a infraestrutura Walmart entrou em contato com a equipe Vtex. Identificou-se que um dos pontos forte da Vtex era a integração contínua realizada em seu ambiente de nuvem. Após análise, concluiu-se que a possível solução para os problemas enfrentados seria a interligação deste processo de integração ao ambiente Walmart.

Como a área de negócio estava demandando um retorno rápido do investimento, eram necessárias medidas simples e ágeis. Para viabilizar a solução, do ponto de vista de segurança e interligação de redes, a equipe do Walmart forneceu à Vtex uma Virtual Private Network (VPN) numa VLAN separada com *proxy* dedicado.

Considerou-se que a solução deveria dar visibilidade da infraestrutura disponível à Vtex, mas sem expor os elementos de outras soluções. Como a Walmart trabalhava

com o conceito de Infraestrutura como Código – viabilizado através da ferramenta Chef⁴. – e todo o código de infraestrutura já estava disponível em um sistema de controle de versão (Git⁵), para atender a este requisito optou-se por realizar uma reestruturação de seus módulos, criando um submódulo específico para a infraestrutura do *marketplace* Vtex.

Para suportar a implantação de versões de forma ágil, foram disponibilizadas soluções utilizando os sistemas de automação – Jenkins⁶ e Chef – já utilizados no Walmart. Os desenvolvedores selecionavam as versões que deveriam ser implantadas e podiam executar através de telas simples e práticas ações como leitura e exclusão de logs, visualização de eventos, reinicialização de servidores de aplicação, busca de cadeia de texto e comparação de arquivos.

A solução implantada para a Vtex obteve sucesso e foi incorporado ao modelo de desenvolvimento da Walmart. Cada equipe de desenvolvimento tinha visibilidade da infraestrutura através de um submódulo do Git, utilizava as ferramentas disponíveis no Chef e Jenkins tanto para depurar o código quando para realizar a integração contínua até o ambiente de qualidade. Após a empresa contratada pela Walmart para realizar o processo final de homologação aprovar a versão, a aplicação era disponibilizada em produção. Ao negócio Walmart eram realizadas entregas mais ágeis e mais atualizadas e sincronizadas com relação à versão em nuvem.

2.2 AGCO: da fábrica ao DevOps

Lucas (2015) descreveu o caso da AGCO⁷, uma empresa de grande porte, multinacional do ramo de máquinas agrícolas. Em 2012 o trabalho de desenvolvimento era orientado à fábrica de software. Na avaliação de Lucas (2015), a área de TI da companhia era um expectador entre o negócio e os fornecedores externos de solução e desenvolvimento. A empresa não tinha propriedade intelectual das soluções e nem possuía as habilidades técnicas necessárias. As entregas em produção eram

⁴ Site oficial: <https://www.chef.io/>

⁵ Site oficial: <https://git-scm.com/>

⁶ Site oficial: <http://jenkins-ci.org/>

⁷ Site oficial: <http://www.agcocorp.com/>

complexas, incluindo documentos de dezenas de páginas. Os desenvolvedores da fábrica não tinham garantias do código que estava sendo executado em ambiente produtivo e, por isto, solicitavam inclusive alterações específicas em linhas do código fonte das páginas Web disponíveis no servidor (ao invés de substituir o arquivo com uma versão mais recente).

Neste cenário, o tempo médio para montagem de um ambiente não produtivo chegava a mais de um mês, o tempo médio para implantação era de 12 horas e a atualização de um ambiente não produtivo para se aproximar do ambiente de produção era realizada quatro vezes por ano.

Em 2013, a organização decidiu focar em *ownership*. Requisitou-se a propriedade intelectual das soluções desenvolvidas, os códigos fontes passaram a residir no sistema de controle de versão da empresa e os processos de implantação em produção passaram à gestão interna. O apoio da alta administração foi indispensável durante a negociação com os fornecedores.

Este movimento de aquisição de *ownership* começou na divisão da América do Sul, onde se introduziu as ferramentas Bitbucket⁸, Jira⁹, Jenkins e Sonarqube¹⁰. Nesta etapa, eliminou-se um ambiente extra de pré-produção e conseguiu-se baixar pela metade (seis horas) o tempo necessário para uma implantação. Lucas (2015) considera que faltou em 2013 um melhor planejamento das ações, além de se ter fornecido uma liberdade excessiva com relação à configuração e acessos do ambiente. Para ele, neste ano também houve muita customização de scripts o que tornava difícil a gestão.

Em 2014, focou-se em tornar a experiência da América do Sul num modelo global. Optou-se por manter a simplicidade, reduzindo o número de configurações e personalizações. Decidiu-se por privilegiar as soluções de automatização de um único fornecedor, usufruindo-se das integrações nativas entre as soluções. Incorporou-se as

⁸ Site oficial: <https://bitbucket.org/>

⁹ Site oficial: [https://www.atlassian.com/software/jira/#!](https://www.atlassian.com/software/jira/)

¹⁰ Site oficial: <http://www.sonarqube.org/>

ferramentas da Atlassian¹¹ (Bamboo¹², Stash¹³, Confluence¹⁴) além do flyway¹⁵, GitHub¹⁶, Travis CI¹⁷ e heroku¹⁸.

Como resultado desta etapa, todos os ambientes não produtivos passaram a ter implantação automatizada. Em produção, segundo Lucas (2015), por resistência e medo da equipe de operações, ainda existe um processo de avaliação e execução manual do script de implantação. O tempo médio de montagem de um ambiente não produtivo passou a duas horas e o tempo médio de implantação em produção passou a uma hora (incluindo cerca de 40 minutos para backup do banco de dados).

As resistências encontradas com os fornecedores na América do Sul tomaram amplitude global em 2014. O *know-how* em DevOps e automação autodeclarado no momento da contratação, muitas vezes não era percebido na prática. Outro ponto de atenção, de acordo com Lucas (2015), é que, neste momento, alguns conhecimentos específicos do funcionamento das ferramentas e soluções de DevOps utilizadas dentro da AGCO ainda estavam restritos a alguns membros da equipe. Havia necessidade de melhorar a gestão do conhecimento.

Em 2015, com o ganho de *ownership*, o objetivo é conhecimento das soluções implantadas na empresa, tendo consciência de seu funcionamento. Há um processo de introdução de novas ferramentas como Docker¹⁹, New Relic²⁰, logentries²¹, coveralls²², mesosphere²³. As práticas de monitoração estão sendo reavaliadas, pois as que atendem à infraestrutura não atendem às demandas de aplicação. Existe uma expectativa de redução de tempo de disponibilização de um novo ambiente não produtivo para cinco minutos e o tempo de implantação pode reduzir para um minuto.

¹¹ Site oficial: <https://www.atlassian.com/>

¹² Site oficial: [https://www.atlassian.com/software/bamboo/#!](https://www.atlassian.com/software/bamboo/)

¹³ Site oficial: <https://www.atlassian.com/software/stash/>

¹⁴ Site oficial: <https://www.atlassian.com/software/confluence/#!>

¹⁵ Site oficial: <http://flywaydb.org/>

¹⁶ Site oficial: <https://github.com/>

¹⁷ Site oficial: <https://travis-ci.org/>

¹⁸ Site oficial: <https://www.heroku.com/>

¹⁹ Site oficial: <https://www.docker.com/>

²⁰ Site oficial: <http://newrelic.com/>

²¹ Site oficial: <https://logentries.com/>

²² Site oficial: <https://coveralls.io/>

²³ Site oficial: <https://mesosphere.com/>

Para Lucas (2015), as soluções de DevOps precisam considerar o contexto e as necessidades locais, assim como elementos culturais e tecnológicos. É necessária uma estratégia de mudança gradual, que entregue valor a cada ação, difundindo o conhecimento pelas equipes.

2.3 CI&T: DevOps na nuvem

Viveiros (2015) relata o caso da CI&T²⁴. Para ele, DevOps é a união das equipes de desenvolvimento e operações para melhorar a comunicação, colaboração e integração. Inicialmente o time de tecnologia era composto por cinco desenvolvedores. Optou-se por não trabalhar com uma equipe de operação local, sendo o suporte de infraestrutura e operações delegado à plataforma de nuvem da Google – *Google Cloud Platform*²⁵. A estratégia montada focava no desenvolvimento, na agilidade (implantando códigos para produção o mais rápido possível) e qualidade (considerando solução, performance e escalabilidade). Nesta direção, a equipe de TI da organização, trabalhava integralmente na construção do produto, utilizando soluções de nuvem – como o PaaS – que oferecessem um nível de abstração das atividades de operação de infraestrutura.

Com o uso do serviço de nuvem, segundo Viveiros (2015), a CI&T conseguiu uma infraestrutura pronta para uso, homogênea entre os ambientes (desenvolvimento, qualidade e produção), escalável e com alta disponibilidade. Na nuvem também era ofertado serviço de controle de versões implantadas, possibilitando um retorno ágil às versões anteriores.

As ferramentas internas da CI&T foram integradas à nuvem. O Jenkins, ferramenta utilizada para integração contínua, realizava as implantações no ambiente da nuvem das versões dos códigos que eram gerenciadas pelo Bitbucket o qual estava integrado com o Jira Cloud que realizava o gerenciamento/documentação dos problemas e melhorias para a solução.

Para reduzir o risco das implantações em produção, a CI&T adotou uma estratégia de implantar as novas funcionalidades desabilitadas. Uma vez implantada a nova versão,

²⁴ Site oficial: <http://www.ciandt.com/>

²⁵ Site oficial: <https://cloud.google.com/>

as funcionalidades eram habilitadas primeiramente nos sites da própria empresa e depois, caso não fossem detectados problemas, nos sites dos seus clientes. Esta estratégia demanda um esforço no desenho de arquitetura para considerar a habilitação seletiva de funcionalidades.

O monitoramento realizava-se através das próprias ferramentas da nuvem que exibiam dados como quantidade de erros, instâncias ativas e requisições por endereço. Alertas podiam ser configurados a determinados eventos, disparando e-mails de aviso quando necessário.

Ao passar a responsabilidade da infraestrutura para um serviço da nuvem no modelo PaaS, a CI&T precisou tratar e acompanhar o custo da operação. Necessitava-se conhecer as formas como serviço era cobrado e acompanhá-lo de forma a evitar imprevistos que podiam desestabilizar a empresa economicamente. Através das informações providas pelo próprio serviço de nuvem e analisadas pela empresa, era possível identificar o custo de cada ambiente (produtivo ou não) e os elementos que compunham este custo. Com estas informações, a equipe de desenvolvimento pôde trabalhar em códigos que tivessem uma relação custo/benefício mais favorável.

A evolução da arquitetura do produto da CI&T demandou um conjunto de funcionalidade não disponíveis na nuvem no modelo PaaS. Para atender a esta necessidade, mas mantendo o modelo de negócio em nuvem, optou-se, para esta parte específica da solução, por um serviço no modelo Infraestrutura como Serviço (*Infrastructure as a Service – IaaS*). Desta forma, a aplicação da empresa funcionava em sua maior parte no modelo PaaS com uma integração localizada no modelo IaaS no qual a complexidade relativa à montagem e manutenção dos servidores era abstraída pela nuvem. Como a parte da solução que necessitava da IaaS era uma execução em lote, a CI&T podia ligar e desligar os servidores quando necessário, controlando os custos de infraestrutura.

O uso de *containers* começou a ser realizado com o crescimento da equipe para prover os ambientes de desenvolvimento com todas as ferramentas necessárias em um curto

espaço de tempo. Todo o código dos containers era controlado pelo GitHub e a ferramenta utilizada para *containers* foi o Docker.

2.4 Zendesk: DevOps com equipes geodistribuídas

Aquino e Saito (2015) relatam o caso da Zendesk²⁶, uma empresa multinacional que vem crescendo rapidamente através da oferta de um sistema de atendimento (tickets) no modelo *Service as a Service* (SaaS) que trabalha com múltiplos canais de interação (redes sociais, email, telefone, sistemas de helpdesk etc.).

Com o crescimento rápido da empresa, a infraestrutura teve que se adequar e criar soluções para entregar performance e escalabilidade da solução. Para atender à esta demanda, adotou-se uma solução de pontos de distribuição (*Point of delivery* – PoD). Cada cliente é alocado em um PoD o qual é autossuficiente contendo uma versão completa da solução de atendimento da Zendesk. Com esta estratégia, como os PoDs estão geograficamente espalhados pelo globo, é possível distribuir os clientes de forma a entregar uma melhor performance. Por outro lado, como cada PoD é autossuficiente, o domínio dos incidentes é restrito, ou seja, um problema em determinado PoD não afeta todos os clientes. A orquestração dos PoDs é gerida pela ferramenta Chef de forma centralizada, porém, para reduzir os riscos de implantação em produção, os PoDs não recebem uma nova versão do software ao mesmo tempo.

O Chef se insere nos processos da Zendesk que são baseados em *Information Technology Infrastructure Library* (ITIL). As requisições de mudança podem vir através de projetos (planejados a cada trimestre) e requisições (tarefas pequenas para o desenvolvimento ou operações). O papel do Chef é atender à gerência de configuração e é tratado pela equipe de DevOps. Toda mudança de configuração exige um ticket ligado a uma alteração no Git e deve ser revisada por outro membro da equipe (*peer review*). Necessariamente a mudança precisa ser comunicada, para isto utiliza-se a ferramenta Flowdock²⁷. Uma vez versionada e comunicada, a mudança é propagada através do Chef.

²⁶ Site oficial: <https://www.zendesk.com/>

²⁷ Site oficial: <https://www.flowdock.com/>

Mudanças médias e grandes que possuem previsão ou potencialmente podem provocar algum tipo de indisponibilidade necessitam ser aprovadas pelo gerente e precisam ter um plano de execução e recuperação. Nestes casos, todos os *stakeholders* devem ser comunicados por e-mail e sua execução deverá ser incluída no calendário de mudanças gerido pela ferramenta Google Calendar²⁸.

As mudanças na aplicação são executadas via ferramenta samson²⁹ (frontend Web desenvolvido internamente para o Capistrano³⁰), geralmente pelos desenvolvedores e são comunicadas à equipe de operações. Como os serviços são desacoplados, cada equipe de desenvolvimento pode implantar sua aplicação sem interferir nas demais.

Na gestão dos incidentes é utilizada a ferramenta PagerDuty³¹. Toda a comunicação entre as equipes de suporte, operações e engenharia é feita através do Flowdock. Como a empresa tem uma política de transparência, o Flowdock é integrado ao twitter³² e facilita a divulgação de status a respeito do incidente e a recepção de feedbacks dos clientes. Ao final de cada incidente é requisitada a elaboração de um sumário que será divulgado internamente com informações do problema e sua solução, servindo de instrumento para a gestão de problemas.

Dentre da rotina da gestão de problemas, ocorre uma reunião semanal onde as equipes de operações, de engenharia e de suporte, que participaram da solução dos incidentes da última semana, tratam sobre três tópicos: 1) estatísticas dos últimos noventa dias dos principais fatores de causa dos incidentes – classificados em questões de capacidade, de configuração, de hardware, humanas e de software; 2) análise do pós-incidente (*post-mortem*) com elaboração do sumário dos incidentes da semana com algumas questões chaves (qual foi o impacto e quanto tempo até que o incidente tenha sido resolvido? o que causou o incidente? como a empresa descobriu o problema? existe algum teste para detectá-lo? como foi a resposta e a recuperação?); 3) itens de

²⁸ Site oficial: <http://www.google.com/calendar/>

²⁹ Repositório oficial: <https://github.com/zendesk/samson> (não foi localizado site oficial desta ferramenta).

³⁰ Site oficial: <http://capistranorb.com/>

³¹ Site oficial: <https://www.pagerduty.com/>

³² Site oficial: <https://twitter.com>

remediação, quando é identificado o que pode ser feito para o incidente não se repetir e analisada a probabilidade de nova ocorrência.

Em termos de ferramentas, no tratamento dos incidentes, é utilizado o Github para controle das versões das modificações implementadas. Os testes são realizados com o auxílio do Travis CI, Rspec³³ e Serverspec³⁴. A monitoração de infraestrutura é realizada pela ferramenta Icinga³⁵, mas também são utilizadas o Datadog³⁶ (para métricas instrumentadas manualmente), New Relic (instrumentalização automática através de agentes), o AirBreak³⁷ (facilita relacionamento de problemas com implantações), pingdom³⁸ (monitoração de disponibilidade geodistribuída testando URLs e seus tempos de acesso), ThousandEyes³⁹ (monitora disponibilidade, cobrindo uma área diferente do pingdom e conseguindo também monitorar através de agentes problemas entre PoDs). Para mitigação de segurança são utilizadas o CloudFlare⁴⁰ e o Prolexic⁴¹. Todas as ferramentas de monitoração abrem tickets no PagerDuty.

Aquino e Saito (2015) destacam alguns pontos culturais importantes da Zendesk no que diz respeito à DevOps. Para eles, existe uma proximidade entre a equipe de desenvolvimento e operações, trabalhando como uma única equipe e evitando problemas de comunicação. Cada região geográfica possui estas duas equipes e duas vezes por ano todo o time é reunido em um único local, fora do escritório, para um evento de integração (*team building*) e treinamento. De acordo com Aquino e Saito (2015), estes eventos são fundamentais para criar a confiança. Também na linha de integração e comunicação, semanalmente às segundas são realizados os *Tech talks*, quando as equipes apresentam na sua região geográfica os projetos que estão trabalhando. Nas quartas-feiras não é permitido o agendamento de reunião e as equipes podem trabalhar remotamente. Entretanto, o trabalho remoto também pode

³³ Site oficial: <http://rspec.info/>

³⁴ Site oficial: <http://serverspec.org/>

³⁵ Site oficial: <https://www.icinga.org/>

³⁶ Site oficial: <http://www.datadoghq.com/>

³⁷ Site oficial: <https://airbrake.io/>

³⁸ Site oficial: <https://www.pingdom.com/>

³⁹ Site oficial: <https://www.thousandeyes.com/>

⁴⁰ Site oficial: <https://www.cloudflare.com/>

⁴¹ Site oficial: <http://www.prolexic.com/>

acontecer fora deste dia de acordo com cada caso, pois esta prática é adotada dentro da política organizacional.

A cultura OpenSource também faz parte das práticas organizacionais e, conforme Aquino e Saito (2015) destacam, estimula o desenvolvimento de uma maior qualidade. Os desenvolvedores são incentivados a escreverem seus códigos pensando que eles podem ser tornados públicos, o que supostamente encoraja uma melhor qualidade de código, devido à possibilidade de maior visibilidade do seu trabalho.

Os casos selecionados exemplificam diferentes usos das práticas de DevOps em contextos distintos. Walmart, AGCO, CI&T e Zendesk enfrentaram seus próprios desafios, utilizaram estratégias diferentes e chegaram a soluções distintas. O capítulo seguinte tratará da análise de forma individual e cruzada, identificando as contribuições para a área de estudo.

CAPÍTULO 3 - VALOR PARA O NEGÓCIO, VISÕES E SOLUÇÕES

Para atingir os objetivos deste trabalho, optou-se por sistematizar a análise nos seguintes tópicos (questões chaves): concepção de DevOps, estímulo inicial, principais ferramentas utilizadas e resultados alcançados. O Quadro 2 abaixo possibilita uma visão comparativa do que será abordado em detalhes neste capítulo.

	Concepção de DevOps	Estímulo inicial	Principais ferramentas utilizadas	Resultados alcançados
Caso Walmart	Comprometimento e objetivo comum.	Desatualização, dificuldade de homologação e demora na disponibilização da solução para o negócio.	Git, Jenkins e Chef	Estabelecimento de um novo modelo de desenvolvimento, versões atualizadas e disponibilizadas para o negócio.
Caso AGCO	Junção de desenvolvimento e operações, solução contextual, cultural e tecnológica.	Ausência de propriedade intelectual das soluções, implantações arriscadas e demoradas.	Bitbucket, Jira, Jenkins, Sonarqube, Bamboo, Stash, Confluence, flyway, GitHub, Travis CI e heroku	Ganho de ownership, implantações automatizadas e mais frequentes.

	Concepção de DevOps	Estímulo inicial	Principais ferramentas utilizadas	Resultados alcançados
Caso CI&T	União das equipes de desenvolvimento e operações para melhorar a comunicação, colaboração e integração	Equipe de TI enxuta, foco no desenvolvimento da solução.	Google Cloud Platform, Bitbucket, Jenkins, Jira, GitHub, Docker	Escalabilidade, controle de custos de operação.
Caso Zendesk	Proximidade entre a equipe de desenvolvimento e operações, trabalhando como uma única equipe e evitando problemas de comunicação.	Integração das equipes para modelagem da solução.	Chef, Github, Flowdock, Google Calendar, Samson, Capistrano, PagerDuty, Travis CI, Rspec, Serverspec, Icinga , Datadog, AirBreak, pingdom, ThousandEyes	Integração para resolução de problemas e construção da solução.

Quadro 2: Comparativo entre os casos analisados no que diz respeito à concepção de DevOps, estímulo inicial, principais ferramentas utilizadas e resultados alcançados.

3.1 Concepção de DevOps

DevOps não é um conceito meramente acadêmico. Ele não teve sua origem em uma universidade, mas sim no mercado, em um ambiente dominado pela frustração por não conseguir realizar entregas mais rápidas ao negócio. A TI não conseguia responder com a agilidade necessária e trabalhava em duas equipes em constante conflito.

Realizar dez implantações ao dia não era comum, pois exigia um nível de automação e integração entre as equipes de desenvolvimento e operações geralmente não encontrado nas empresas. Além disto, pelas condições existentes, uma implantação envolvia um risco relativamente alto para o negócio. Por exemplo, a solução poderia ficar indisponível e, em muitos casos, isto representa um produto que não é entregue ao cliente.

Para atender à dinamicidade exigida pelo mercado, encaminhavam-se várias iniciativas de TI, tanto por parte do desenvolvimento – ex.: Métodos Ágeis de Desenvolvimento –, quanto da área de operações – ex.: Infraestrutura como Código. Trabalhando em silos, em um clima de desconfiança e sem métricas compartilhadas, as duas equipes isoladas nem conseguiam funcionar como uma TI eficiente para o negócio, nem entregar as soluções necessárias.

O caso Walmart foi relatado por Lorieri (2015), membro da equipe de infraestrutura e que participou ativamente da solução. Ele percebeu que, apesar da infraestrutura operacional e estável, as entregas não estavam acontecendo ao negócio. Após obter patrocínio da sua gerência, efetuou um movimento de aproximação com a equipe de desenvolvimento. Buscou entender os problemas e as dificuldades, trabalhando junto com a equipe para superar obstáculos.

Lorieri (2015) relaciona DevOps a comprometimento e objetivo comum. Entretanto, enquanto a área de desenvolvimento e a área de operações trabalham com métricas distintas e não relacionadas, provavelmente haverá dificuldade em encontrar um objetivo em comum, pois a cobrança gerencial direcionará para outro caminho – a formação de silos. Neste cenário de silos, no qual cada equipe é cobrada e medida como entidade distinta e apartada, favorece-se o surgimento de incoerências

organizacionais. Por exemplo, as duas equipes podem atingir completamente suas metas, enquanto a TI frustra a organização e atua como um gargalo para as metas do negócio.

O objetivo comum precisa ser organizacional. Na linha dos princípios fundamentais de DevOps, trata-se de focar no valor para o negócio que é criado pela TI, pensando no sistema e não em silos. Ao perder este foco, tem-se uma falsa sensação de dever cumprido. Cada equipe acredita que realizou sua parte, contribuindo para manter os indicadores da sua área dentro dos limites estabelecidos, mas esquece do sistema organizacional e do necessário comprometimento com a estratégia do negócio.

Conforme Zanelli e Silva (2012, p. 90), “[...] comprometimento organizacional significa o grau de identificação de uma pessoa com os valores e objetivos da organização e está associado a uma disposição de trabalhar em prol da organização”. Assim, comprometimento pressupõe que os empregados estejam cientes dos valores e dos objetivos organizacionais. Por outro lado, ainda segundo Zanelli e Silva (2012), a identificação com estes valores e objetivos é favorecida quando o empregado conhece, entende e legitima o que se espera dele.

Desta forma, colaboram para o comprometimento e a construção de um objetivo comum uma gestão transparente – que esclareça qual o papel de cada equipe e empregado dentro do sistema organizacional – e um processo de comunicação eficiente – que informe adequadamente o corpo funcional os valores e objetivos organizacionais. Em âmbito geral, o objetivo em comum da área de TI é entregar soluções demandadas pelo negócio. Desenvolvedores e operadores, comprometidos organizacionalmente, utilizando suas competências específicas, precisam concentrar esforços no valor para o negócio, que só é realizado quando a solução está em funcionamento no ambiente produtivo.

Este comprometimento organizacional conjunto está na linha da conceituação de DevOps apresentada no caso AGCO. Para Lucas (2015), que relatou este caso, DevOps trata da junção de desenvolvimento e operações numa solução contextual, cultural e tecnológica.

No que diz respeito à junção de desenvolvimento e operações, alguns autores, especialmente Hüttlermann (2012), destacam que não se trata de acabar com a área de operações nem com a área de desenvolvimento. Cada uma continuará com suas responsabilidades. O movimento de DevOps não indica que desenvolvedores trabalhem diretamente no ambiente produtivo nem que operadores assumam as atividades dos desenvolvedores. Operadores e desenvolvedores possuem prioridades e responsabilidades diferentes que são necessárias no ambiente organizacional. Para Hüttlermann (2012), DevOps envolve disciplina, convenções e um processo definido que é transparente para todos.

O contexto, a cultura e a tecnologia estão imbricados nas práticas organizacionais e devem ser considerados na adoção de processos e soluções. No caso AGCO, por exemplo, trabalhava-se com fábricas de software, o que não acontecia com a CI&T, a qual desenvolvia sua própria solução com uma equipe de desenvolvimento local. Contextos diferentes exigem abordagens diferentes, não sendo possível falar em uma forma única para aplicar as práticas de DevOps na organização.

Criada coletivamente, a cultura organizacional sofre mudanças não somente devido à alteração ambientais, mas também como resultado das ações de gestão. A visão gerencial dominante e a cultura organizacional estão estreitamente relacionadas. O movimento da mudança é um esforço para alterar as formas vigentes de cognição e ação, possibilitando que a organização se adeque às condições ambientais e aproveite as oportunidades que se apresentam. (VASCONCELOS, 1993)

As mudanças culturais são motivadas por modificações ambientais e pela necessidade de adaptação. No caso da AGCO, percebe-se que a ausência de propriedade intelectual das soluções, junto com a demora em realizar implantações, representava um risco à sobrevivência organizacional. Em certa medida, o conhecimento das soluções de TI estava sendo terceirizado. A visão gerencial direcionou uma mudança na cultura organizacional. Na TI, a necessidade de se ter propriedade intelectual das soluções passou a ser incorporada aos valores organizacionais. Isto, por sua vez, moldou uma mudança tecnológica, pois novas ferramentas foram incorporadas ao processo de forma que atendessem ao novo contexto cultural.

Os elementos contextuais, culturais e tecnológicos devem ser considerados ao avaliar as práticas de DevOps que serão adotadas. A diferença destes elementos nos casos apresentados é bastante emblemática e influencia no conceito de DevOps apresentado pelo relator de cada caso.

A CI&T trabalhava na nuvem com uma equipe de desenvolvimento própria. Para Viveiros (2015), DevOps é a união das equipes de desenvolvimento e operações para melhorar a comunicação, colaboração e integração. Cabe destacar, entretanto, que a CI&T não tinha, no seu corpo funcional, uma equipe de operações, este papel era inteiramente desempenhado pelo fornecedor da nuvem. Contexto, cultura e tecnologia eram bem distintos dos dois casos anteriores (Walmart e AGCO).

O fornecedor da nuvem disponibilizava à CI&T uma série de ferramentas que possibilitavam a gestão das entregas de software, o acompanhamento de métricas e a gestão do ambiente. Durante a descrição do caso, não foi feita qualquer referência à interação direta com as pessoas da equipe de operações da nuvem e, conforme os padrões de mercado do modelo de negócio de nuvem, é muito provável que a equipe de desenvolvimento interna da CI&T nunca teve qualquer contato direto com a equipe de operadores.

Neste caso relatado por Viveiros (2015), há uma peculiaridade interessante: afirma-se que se utilizou prática de DevOps quando todos os elementos levam a crer que as pessoas das equipes de desenvolvimento e operações sequer se conheciam. Ou seja, há um aparente contrassenso entre a definição de DevOps apresentada pelo próprio Viveiros (2015) e o caso relatado. Entretanto, analisando-se as estratégias tecnológicas adotadas de acordo com os princípios fundamentais de DevOps, percebe-se que as ferramentas tecnológicas disponibilizadas na nuvem permitiam uma aproximação entre desenvolvedor e fornecedor de operações.

Apesar disto, não é possível afirmar que o fornecedor da nuvem estava comprometido com os valores e objetivos da CI&T, mas que a estratégia deste estava alinhada às necessidades da CI&T. A aproximação entre desenvolvedores e operações foi mediada

pelas tecnologias e deu-se no compartilhamento de valores estratégicos no âmbito da TI.

Comunicação, colaboração e integração eram facilitadas pelas ferramentas disponíveis. As práticas de implantação, manutenção da disponibilidade e custo são exemplos relevantes neste sentido. Implantações podiam ser realizadas e desfeitas, inclusive em produção, pela própria equipe de desenvolvimento, sem necessidade de processos burocráticos, utilizando ferramentas automáticas e simples. A disponibilidade era garantida ao negócio por meio de acordos contratuais e a escalabilidade era uma possibilidade, solicitável de acordo com a demanda do negócio. A tarifação da infraestrutura era transparente, podendo ser acompanhada pelos desenvolvedores, desta forma dados da operação realimentavam o desenvolvimento para que códigos mais eficientes e otimizados em termos de custo pudessem ser desenvolvidos.

Na Zendesk, existe tanto uma equipe interna de desenvolvimento quanto uma equipe interna de operações, porém também são utilizadas várias ferramentas que facilitam a aproximação entre elas. Para Aquino e Saito (2015), DevOps diz respeito à proximidade entre a equipe de desenvolvimento e operações, trabalhando como uma única equipe e evitando problemas de comunicação.

Nesta concepção de DevOps, assim como nos casos anteriores, está implícita a importância de se construir uma equipe de TI, com objetivos compartilhados que sirva ao negócio. Um ponto relevante no caso da Zendesk é a preocupação que desenvolvedores e operadores participem em conjunto da solução dos problemas. Percebe-se que há na cultura o entendimento de que um problema em TI precisa ser solucionado com a participação de todos – desenvolvedores e operações – e, para isto, um processo de comunicação eficiente precisa ser construído.

Esta preocupação com a comunicação é respaldada não só nas ferramentas utilizadas, mas na disposição geográfica das equipes pelo globo e nas ações de integração e compartilhamento de informações. Em cada escritório espalhado pelo globo, existem equipes de desenvolvimento e operações que interagem rotineiramente nos projetos e tratamento de problemas. Os eventos de integração bianuais que juntam toda a equipe

da Zendesk e os *Tech talks* são oportunidades de trabalhar as habilidades de comunicação e integração do grupo, favorecendo a construção de um espírito de equipe único dentro da TI.

De modo geral, as diversas concepções de DevOps apresentadas pelos relatores de cada caso, apontam para um mesmo lugar. Comprometimento, objetivo comum, contexto, cultura, tecnologia, comunicação, colaboração e integração devem ser direcionados e trabalhados para que a TI agregue valor ao negócio, entregando as soluções de forma ágil e confiável de maneira a reduzir o ciclo de feedback. Desenvolvedores e operadores possuem suas próprias habilidades e especialidades, mas devem trabalhar para o negócio como uma equipe de TI única, corresponsável pelos resultados organizacionais. Para isto, não há uma fórmula a ser aplicada ou padrão único a ser utilizado. A solução é contextual e o estímulo para iniciar o movimento de DevOps pode ser bastante distinto.

3.2 Estímulo inicial

Quando se fala do estímulo para iniciar o movimento de DevOps, a referência é algo que provocou um comportamento de mudança. Ou seja, trata-se da construção ou da reconstrução de uma identidade e de uma forma de trabalhar.

Na contemporaneidade, cada vez mais os funcionários são convidados a vestir a camisa da empresa, o que exige consciência e engajamento dos valores organizacionais e clara definição da estratégia e dos objetivos do negócio. O que é desejável ou não em cada contexto é introjetado pelos indivíduos de acordo com as interações cotidianas. Desta forma, os funcionários são conduzidos para uma uniformidade de atitudes, o que é positivo no que diz respeito à coesão do grupo, porém pode levar a uma perda de individualidade, pois o comportamento do indivíduo passa a ser uma extensão do grupo. (FERNANTES, ZANELLI; 2006)

“[...] identidades precisam ser reconstruídas, quando a empresa se vê diante de situação que exigem mudanças” (FERNANTES, ZANELLI; 2006, p. 57). Durante o processo de reconstrução, experiências positivas de outras comunidades podem ser incorporadas, cada indivíduo constrói ou reconstrói sua identidade através de múltiplos

fatores, dentre os quais as estruturas sociais, a cultura e o histórico das relações. Daí a importância de incentivar a participação em treinamentos externos, comunidades e eventos que possibilitem a oxigenação da cultura organizacional.

Tanto no caso Walmart quanto no caso AGCO houve uma necessidade de reconstrução da identidade corporativa. Para o Walmart, isto foi mais discreto, no âmbito de apenas um projeto específico – porém importante – que depois teve seus frutos incorporados às práticas organizacionais no Brasil. Na AGCO, as mudanças fizeram partes de um novo direcionamento corporativo.

Em âmbitos diferentes, estes dois casos tiveram como estímulo inicial o problema de entrega de soluções de TI. O valor não estava sendo entregue ao negócio. Os ciclos de feedback estavam longos, pois decorria um longo tempo desde que uma alteração era efetuada no software até o momento de sua disponibilização em produção para que o negócio pudesse colher os resultados e aprender com a experiência.

No Walmart, a solução começou a ser desenhada pela equipe de infraestrutura, que percebeu que de nada adianta a sua parte estar operacional, mas o software suportado não gerar valor. É interessante perceber neste caso que a Vtex, empresa responsável pelo desenvolvimento, conseguia entregar adequadamente sua solução em formato de nuvem. O problema estava acontecendo especificamente nas entregas ao ambiente Walmart. O importante trabalho iniciado pela infraestrutura foi entender e ultrapassar barreiras. Precisou-se compreender o que estava falhando no processo e traçar estratégias de superação.

Na AGCO, as soluções eram todas desenvolvidas pelas fábricas com base nas especificações da equipe de arquitetura. Os fornecedores entregavam as soluções, mas não tinham conhecimento ou segurança daquilo que realmente estava em execução em ambiente produtivo (infraestrutura ou código), o que tinha como consequência complexos documentos de implantação com alto risco de falha e com um tempo longo de execução.

Coube à equipe interna de arquitetura o desafio de se mobilizar em busca da solução. Em um processo de reaquisição da propriedade intelectual de suas soluções, com

apoio da alta direção, negociou com as fábricas e modificou a forma de trabalho, criando padrões de repositório e disponibilizando ferramentas que facilitavam a entrega em ambiente não produtivo. Gradualmente, a equipe de arquitetura conseguiu aproximar a fábrica da equipe de operações da AGCO.

Diferente da Walmart e da AGCO, a CI&T e a Zendesk foram concebidas direcionadas ao DevOps. No caso da CI&T, o estímulo para as práticas de DevOps foi entregar uma solução robusta utilizando todo o pessoal disponível localmente para desenvolvimento. Assim, a infraestrutura terceirizada (nuvem) precisava disponibilizar mecanismos que possibilitassem uma entrega ágil das soluções de TI, mas que também oferecesse a disponibilidade e escalabilidade adequadas.

Observe-se que a decisão da CI&T foi estratégica. Ela preferiu focar seus recursos internos no desenvolvimento de sua solução a qual, em última instância, era o produto que tinha a oferecer ao mercado. A terceirização ao modelo em nuvem delegou a uma empresa de renome a responsabilidade de disponibilizar sua solução aos seus clientes. Para o sucesso, foi essencial a adequação nos meios de comunicação, pois sem ela as entregas ao negócio seriam prejudicadas.

A Zendesk incorporou DevOps desde o início, mas com o estímulo e foco na integração das equipes para modelagem da solução. Da forma como o caso foi descrito por Aquino e Saito (2015), parece que o objetivo foi construir uma equipe de TI que, embora contemplasse os papéis de desenvolvedores e operadores, atuasse em conjunto desde o desenho da solução até à resolução de incidentes. As práticas de DevOps estão perfeitamente alinhadas a este direcionamento no qual a integração entre as pessoas e o processo de comunicação são elementos centrais.

Assim como na Zendesk, em diferentes níveis, pode-se perceber que nos quatro casos, a integração entre pessoas e processos foi fundamental. O estímulo inicial sempre perpassou pela entrega das soluções ao negócio, mas a estratégia organizacional de DevOps focava em pessoas e processos – principalmente processos de comunicação. As ferramentas incorporadas só tiveram sentido quando inseridas dentro desta estratégia.

3.3 Principais ferramentas utilizadas

Para facilitar a análise, o Quadro 3 abaixo agrupa as ferramentas utilizadas nos quatro casos examinados nas seguintes categorias: gestão de fontes/versões, gestão de estrutura de banco, gestão de infraestrutura, infraestrutura em nuvem, automação, gestão de incidentes/problemas, gestão da comunicação, monitoração e segurança. Embora algumas ferramentas pudessem ser enquadradas em mais de um grupo, optou-se por manter cada aplicação em uma única categoria de acordo com o uso nas empresas analisadas.

Finalidade principal	Ferramentas
Gestão de fontes/versões	Git/GitHub, Bitbucket, Stash
Gestão da estrutura de banco	flyway
Gestão de infraestrutura	Chef, Docker
Infraestrutura em nuvem	Google Cloud Platform, heroku
Automação	Jenkins, Bamboo, Capistrano, Samson
Gestão de incidentes/problemas	Jira, PagerDuty
Gestão da comunicação	Confluence, Flowdock, Google Calendar,
Gestão de qualidade	Sonarqube, Travis CI, Rspec, Serverspec
Monitoração	Icinga, Datadog, pingdom, ThousandEyes, AirBreak
Segurança	CloudFlare, Prolexic

Quadro 3: Ferramentas utilizadas e suas finalidades principais

Em todos os casos, percebeu-se o uso de alguma ferramenta para gestão de fontes/versões. Estas ferramentas podem armazenar o código fonte e todos os itens de configuração de um software, permitindo acesso a versões anteriores,

compartilhamento entre equipes e recuperação de informações sobre alterações (ALMEIDA; SILVA, 2014). É importante observar que, como foi destacado no caso do Walmart e da Zendesk, o controle de versão não é apenas do código da aplicação. Principalmente no contexto de DevOps, o versionamento é também do código de infraestrutura. Na Zendesk, foi relatada a prática de associar as alterações efetuadas a tickets de incidentes e problemas de forma a facilitar a rastreabilidade das alterações.

A AGCO foi o único caso no qual houve relato de uso de uma ferramenta de gestão da estrutura de banco (*database schemas*). Para a maioria das aplicações de negócio, o banco de dados é um elemento importante da solução. No contexto de entrega contínua, onde geralmente se trabalha com diversos ambientes (desenvolvimento, teste, homologação e produção), manter a estrutura da base de dados em sintonia com as várias versões da aplicação é um desafio. Caso não se tenha o devido cuidado, o software poderá caminhar em uma versão e o banco de dados em outra completamente diferente, o que potencialmente representa riscos para o negócio. As ferramentas de gestão da estrutura de banco atuam de forma a facilitar o gerenciamento das versões dos *database schemas*. Elas podem executar tarefas como migração das estruturas em bancos existentes e criação de um banco de dados novo para disponibilização de um novo ambiente.

Walmart, CI&T e Zendesk relataram já utilizar ferramentas de gestão de infraestrutura; a AGCO, quando o caso foi descrito, estava em processo de adoção de uma solução nesta direção. As ferramentas deste grupo possibilitam gerir a infraestrutura de uma aplicação agilizando a entrega dos ambientes, garantindo sua integridade e a compatibilidade entre infraestrutura e aplicação. Elas tanto podem funcionar descrevendo em formato de código a infraestrutura necessária para aplicação (o que permite criar versões da infraestrutura que sejam compatíveis com cada versão da aplicação), quanto podem utilizar soluções de container (elemento lógico que contém a aplicação a ser executada e todas as suas dependências, diferindo de solução de virtualização, pois roda no espaço do usuário diretamente no sistema operacional hospedeiro). Estas ferramentas tanto podem trazer benefícios para os ambientes corporativos compartilhados, quando para os ambientes individuais. A equipe de

operações pode, por exemplo, manter os ambientes de qualidade e produção com as mesmas configurações e a equipe de desenvolvimento pode garantir que todos os seus desenvolvedores estão utilizando as mesmas versões das ferramentas de desenvolvimento.

O grupo infraestrutura em nuvem engloba produtos de IaaS e PaaS. O maior benefício ao se adotar uma destas soluções é a terceirização da gestão da infraestrutura. Preocupações relacionadas à manutenção de hardware e ao desenho da arquitetura de alta disponibilidade e escalabilidade são delegadas ao fornecedor da nuvem. Apenas a AGCO e a CI&T destacaram o uso destas soluções, sendo que a CI&T tem seu modelo de negócio fortemente voltado à nuvem utilizando-a tanto com IaaS quanto com PaaS.

No grupo automação estão as ferramentas responsáveis por automatizar a trilha de implantação (construção, implantação, teste e liberação). Geralmente estas ferramentas interagem com diversas outras e, por isto, possuem algum mecanismo ou dispositivo de integração (ex.: *plugin*). Elas podem orquestrar desde a interação com o sistema de gestão de fontes até à liberação em ambiente produtivo, passando por etapas como testes automatizados, fluxos de aprovação e resolução de dependências. Observou-se o uso das ferramentas deste grupo em todos os casos relatados.

Com exceção do Walmart, em todas as outras empresas foi citada alguma ferramenta para gestão de incidentes/problemas. As soluções deste grupo auxiliam desenvolvedores e operadores a documentarem falhas, correções e melhorias identificadas para um produto. Usualmente associa-se cada incidente ou problema a um número, uma descrição e algumas categorias gerando-se um ticket. Cada ticket é priorizado e tratado pela equipe. A documentação da solução, seu custo, a quantidade de horas gastas, o número de pessoas e equipes envolvidas podem ser inseridos na ferramenta possibilitando análises gerenciais e ao mesmo tempo criando uma base de conhecimento do produto.

No movimento de aproximação das equipes de desenvolvimento e operações, as ferramentas para gestão da comunicação podem ser úteis. O benefício destas soluções é tornar ágil o processo de troca de informações, seja disponibilizando um calendário

de atividades ou através de uma sala virtual na qual desenvolvedores e operadores podem trocar informações sobre a análise de um incidente em produção. Alguns dos produtos deste grupo são designados de plataformas de colaboração, por proverem mecanismos (wiki, espaço para compartilhamento de documentos etc.) para que pessoas com objetivos em comum interajam para atingi-los. O uso das ferramentas deste grupo foi abordado pela AGCO e Zendesk.

As ferramentas agrupadas na categoria gestão de qualidade geralmente possuem integração com os produtos de automação da trilha de implantação. Elas são responsáveis por realizar diversos tipos de teste para garantir que a solução está de acordo com os parâmetros especificados. Do ponto de vista de DevOps e do negócio, o uso destas ferramentas é extremamente importante, pois reduz o risco de que problemas sejam descobertos apenas em ambiente produtivo. Quando maior for a cobertura dos testes, menor será a probabilidade de uma alteração defeituosa – seja de infraestrutura ou da aplicação – ser liberada para os clientes em produção. Tanto no caso AGCO quanto no da Zendesk foi relatado a utilização destas ferramentas.

A monitoração permite que o negócio atue com agilidade para resolver problemas que afetem seu ambiente. Uma vasta quantidade de ferramentas com esta finalidade está em uso pela Zendesk para tentar localizar e reduzir o impacto e a amplitude de um incidente na sua arquitetura geodistribuída. A monitoração pode identificar tanto problemas na infraestrutura (ex.: servidor indisponível) tanto problemas na aplicação (ex.: exceção reportada no log a partir de uma determinada implantação). De forma a prover transparência ao processo, é interessante que tanto desenvolvedores quanto operadores tenham acesso aos dados providos pela monitoração.

Na categoria segurança foram agrupados produtos que têm por objetivo mitigar riscos de segurança e proteger a organização de ataques direcionados a sua infraestrutura e/ou a suas aplicações. Os ataques de *Distributed Denial of Service* (DDoS), por exemplo, podem indisponibilizar aplicações causando sérios prejuízos tanto financeiro quando relacionados à imagem organizacional. Produtos que mitigam os riscos de segurança foram citadas principalmente pela Zendesk.

Independente do arsenal de ferramentas utilizadas, certamente não foram elas as principais responsáveis pelos resultados alcançados no que diz respeito ao valor que a TI conseguiu agregar ao negócio utilizando práticas de DevOps. Como destaca Hüttlermann (2012), para dinamizar a entrega de software, pessoas e processos são mais importantes que ferramentas.

3.4 Resultados alcançados

Cada um dos casos relatados alcançou resultados positivos com o uso das práticas de DevOps que ultrapassam a simples adoção de uma tecnologia específica. Estimulados por diferentes desafios, as equipes de TI citadas construíram ou reconstruíram uma nova concepção de como desenvolvedores e operadores interagem para entregar valor ao negócio.

No caso Walmart, o movimento da equipe de infraestrutura para se aproximar do fornecedor de aplicação e entender as dificuldades teve como resultado o estabelecimento de um novo modelo de desenvolvimento que foi adotado para outros projetos. A solução da Vtex finalmente conseguiu ser mantida de forma atualizada na infraestrutura da Walmart e era entregue ao negócio em menos tempo e com menor risco. A cultura organizacional foi modificada, inicialmente no nível micro, quando a infraestrutura negociou uma aproximação com a Vtex, mas depois no nível macro, quando o sucesso do modelo foi expandido para outros projetos da organização.

Na AGCO, a estratégia corporativa de adquirir a propriedade intelectual das suas soluções, teve como consequência uma mudança cultural na forma como a organização se relacionava com seus fornecedores. Era importante que eles compreendessem e se adequassem ao novo direcionamento organizacional e também conseguissem entregar valor à organização de uma forma mais ágil. Com as práticas de DevOps, as equipes de desenvolvimento na fábrica conseguiram entregar soluções no ambiente operacional da AGCO com maior frequência e utilizando processos automatizados, trazendo maior confiabilidade às soluções de TI e menor risco ao negócio. A cultura como a TI se relacionava com seus fornecedores mudou, incorporou DevOps e transformou um processo de implantação longo e arriscado em uma

atividade simples, com o mínimo de intervenção humana e que reduziu o ciclo de feedback ao entregar soluções em produção com maior frequência.

A CI&T construiu uma cultura focada no desenvolvimento, mas utilizou práticas de DevOps que moldaram seus processos de comunicação com o fornecedor da nuvem e colaboraram para entregar continuamente soluções para o negócio. Desenvolvedores integravam suas ferramentas locais de automação com a nuvem e conseguiam realizar com agilidade suas entregas. Por outro lado, a política transparente de custo é um dos elementos culturais mais relevantes e que demonstra a proximidade e o ciclo de feedback da relação entre desenvolvedores e equipe de operações. As ferramentas de monitoração davam rastreabilidade do custo da operação das soluções desenvolvidas. Com base nestes dados, os gestores do negócio podiam, por exemplo, identificar onde investir para reduzir custos, mas ao mesmo tempo tinham a clareza do que compunha os custos de infraestrutura.

A Zendesk aproximou desenvolvedores e operadores, tanto globalmente quanto em cada uma das suas bases geodistribuídas. Entregava-se ao negócio soluções de TI que tinham um alinhamento entre desenvolvimento e operação. A cultura da responsabilidade compartilhada imbricada nestas entregas refletia no tratamento dos problemas, que não eram vistos como de uma equipe específica, mas como uma questão de TI, que estava afetando o negócio e que precisava de colaboração de todos para ser resolvida. O valor agregado ao negócio descrito neste caso vai além da agilidade na entrega, ele chega ao suporte dos incidentes e problemas da solução em produção.

Em todos os casos descritos, observa-se que os resultados alcançados perpassaram pelos princípios fundamentais de DevOps. Percebeu-se foco no negócio, com ênfase na performance do sistema organizacional como um todo e não na performance de um silo da empresa. Notou-se a troca de informação (ciclo de feedback) entre desenvolvedores e operadores, mesmo que quando representados por fornecedores, de forma que os feedbacks de uma equipe realimentavam o trabalho e melhoravam as entregas ao negócio. Por fim, relatou-se também um movimento contínuo na direção da

inovação, buscando processos e ferramentas mais eficientes, em um estímulo constante à experimentação que potencialmente trouxesse ganhos ao negócio.

As práticas de DevOps que agregaram valor ao negócio foram bastante variadas entre os casos apresentados, porém é possível sistematizá-las da seguinte forma:

- 1) **Desenho colaborativo de soluções:** conforme visto principalmente no caso da Zendesk, quando desenvolvedores e operadores constroem a solução de forma colaborativa, entrega-se ao negócio um produto da equipe de TI que todos se sentem responsáveis e entendem. Esta prática potencialmente agrega valor para o negócio entregando maior estabilidade e robustez.
- 2) **Entrega contínua:** em todos os casos relatados, buscou-se entregar soluções de TI com agilidade e, ao mesmo tempo, com maior confiabilidade e menor risco. Esta prática potencialmente agrega valor para o negócio na medida em que viabiliza uma resposta rápida às demandas do mercado e dinamiza o ciclo de aprendizado corporativo.
- 3) **Colaboração para superar desafios:** como observado em todos os casos relatados, o trabalho como equipe de TI possibilita que desenvolvedores e operadores unam suas habilidades para superar os desafios cotidianos, seja ele um incidente em produção, um desenho de uma nova solução ou a modelagem de um novo processo que auxiliará no desempenho de suas próprias atividades. Esta prática potencialmente agrega valor para o negócio através da agilidade na superação de desafios.
- 4) **Soluções inovadoras para resolução de problemas:** principalmente nos casos da Walmart e da AGCO, percebeu-se que abrir espaço para um pensamento inovador pode favorecer o ambiente organizacional; a cultura é renovada, processos são otimizados e novas ferramentas são incorporadas. Esta prática potencialmente agrega valor para o negócio pela renovação e modernização dos processos e ferramentas, mantendo vivo o fluxo de aprendizado organizacional.

CONCLUSÃO

Este trabalho buscou identificar as práticas de DevOps relatadas na bibliografia especializada que agregaram valor ao negócio das organizações com atuação em território nacional. Esperava-se contribuir através da metodologia de estudo de casos múltiplos para uma tomada de decisão consciente na priorização de investimentos.

No Capítulo 1, após uma contextualização histórica, conceituou-se DevOps como um movimento de aproximação entre desenvolvedores e operadores que nasceu da necessidade de entregar soluções mais rápidas ao negócio com agilidade e confiabilidade. Analisou-se o conflito inerente à atividade destas duas equipes, mostrando os riscos para o negócio que a falta de colaboração entre desenvolvedores e operadores pode ocasionar (ex.: atraso de entregas e dificuldade na resolução de problemas). Localizou-se a Entrega Contínua como um dos objetivos da cultura de colaboração incentivada pelo movimento de DevOps, mas alertou-se que não se trata de simplesmente entregar soluções rapidamente; precisa-se atender requisitos de qualidades de forma a reduzir o risco de cada implantação. Destacou-se que a prática colaborativa imbricada em DevOps, exige uma ressignificação da cultura organizacional, repensando a forma como as soluções são concebidas e suportadas, numa política da corresponsabilidade entre desenvolvedores e operadores. Identificou-se o papel das ferramentas como auxiliares no processo de colaboração, dando suporte a pessoas e processos. Apresentou-se os princípios fundamentais de DevOps, uma proposta para identificar os níveis de maturidade das organizações e indicou-se o valor agregado como uma das formas que o investimento em DevOps pode retornar para organização. Este capítulo contribuiu principalmente para alcançar os objetivos específicos 1) conceituar DevOps contextualizando-o historicamente; e 2) relacionar o uso de DevOps à cultura organizacional;

O Capítulo 2 tratou da síntese das práticas de DevOps relatadas pela comunidade especializada nas organizações com atuação em território nacional. Sintetizaram-se quatro casos de empresas multinacionais com atuação global, porém com escritórios e equipes de TI no Brasil. Cada caso possuía características próprias que o diferenciava

dos demais, pois, além das empresas operarem em segmentos distintos do mercado, o contexto de aplicação das práticas de DevOps foi variado.

O Capítulo 3 contribuiu para alcançar os objetivos deste trabalho com 1) análise das práticas de DevOps; 2) correlação do conceito de DevOps inserido na cultura organizacional aos resultados alcançados pelas suas práticas; e 3) identificação das práticas que agregaram valor ao negócio. A análise, correlação e identificação de práticas foi feita através de quatro tópicos (questões chaves): Concepção de DevOps, Estímulo inicial, Principais ferramentas utilizadas e Resultados alcançados. O primeiro tópico analisou as práticas adotadas na organização de acordo com a concepção de DevOps inserida na cultura da organização descrita em cada caso. O tópico seguinte analisou e identificou as práticas de DevOps adotadas frente ao Estímulo inicial para a mudança. O terceiro tópico analisou as principais ferramentas utilizadas enquanto facilitadoras das práticas de DevOps adotadas. O último tópico finalizou a análise correlacionando o conceito de DevOps aos resultados alcançados em cada caso e identificando de forma sistemática as práticas que agregaram valor ao negócio.

Esta estruturação do estudo possibilitou que o objetivo geral fosse alcançado com a identificação na bibliografia especializada de práticas de DevOps que agregaram valor ao negócio das organizações com atuação em território nacional. A análise de casos possibilitou a investigação do fenômeno DevOps de forma contextualizada, sem deixar de lado os aspectos culturais fundamentais para análise da questão.

Em uma abordagem diferente da adotada neste trabalho, Kim, Behr e Spaffor (2013) decidiram tratar a temática de DevOps de uma maneira inusitada, através do gênero literário de romance. No enredo, Bill, vindo da equipe de operações, precisa lidar com sua promoção inesperada para gerente de TI da *Parts Unlimited*, uma empresa que está perdendo espaço no mercado por não conseguir entregar soluções aos seus clientes. A última esperança parece ser o Projeto Phoenix, uma solução de TI que busca suprir uma demanda do mercado que seus concorrentes têm conseguido atender e tirar proveito. Infelizmente para Bill, a organização não possui só este projeto, as demandas não param de chegar, desenvolvedores e operadores não se entendem, o processo de mudanças é frequentemente boicotado e cada equipe coloca a culpa nas

outras pelos constantes atrasos do Phoenix. O caso fictício narrado por Kim, Behr e Spaffor (2013) é inspirado em situações reais. O leitor que trabalha com TI certamente já vivenciou algumas das situações descritas.

O romance citado tem em comum com este estudo a temática de DevOps e a possibilidade de identificação do leitor com as situações relatadas. Certamente, entretanto, cabe destacar uma das limitações deste trabalho: as práticas descritas, embora tenham agregado valor para os negócios nos casos abordados, não podem ser generalizadas e adotadas como fórmula descontextualizada e desconectada da cultura organizacional. Ao invés de fórmulas, este estudo contribui principalmente para perceber que DevOps é um movimento que trabalha com pessoas, processos e ferramentas e está fortemente entrelaçado com o contexto cultural da organização. As evidências da literatura e dos casos relatados indicam que DevOps pode sim contribuir para agregar valor ao negócio, desde que suas práticas sejam aplicadas de forma contextualizada e consciente.

REFERÊNCIAS

ALMEIDA, Alicenaira Lanes Souza Carneiro de; SILVA, Evaldo de Oliveira da. **Entrega contínua**: um estudo de caso para automatização do fluxo de implantação do sistema integra. **Caderno de Estudos em Sistemas de Informação**. v. 1, n. 1, 2014. Disponível em <<http://seer.cesjf.br/index.php/cesi/article/view/120/40>>. Acesso 25 de jul. 2015.

AQUINO, Cassiano; SAITO, Eduardo. **DevOps na Zendesk**: desafios e lições sobre arquiteturas escaláveis, processos ágeis e equipes multiculturais. [S.l.]: QCon, 2015. 1 vídeo de apresentação em evento; 1 slide. Disponível em <<http://www.infoq.com/br/presentations/devops-na-zendesk>>. Acesso em 18 de jul. 2015.

COCKCROFT, Adrian. **Adrian Cockcroft's Blog**: Ops, DevOps and PaaS (NoOps) at Netflix., 2012. Disponível em <<http://perfcap.blogspot.com.br/2012/03/ops-devops-and-noops-at-netflix.html>>. Acesso em 23 out. 2014.

CRESWELL, John W. **Investigação qualitativa e projeto de pesquisa**: escolhendo em cinco abordagens. Porto Alegre: Penso, 2014.

FERNANDES, Karina Ribeiro; ZANELLI, José Carlos. O processo de construção e reconstrução das identidades dos indivíduos nas organizações. **Rev. adm. contemp.**, Curitiba , v. 10, n. 1, p. 55-72, mar. 2006 . Disponível em <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1415-65552006000100004&lng=pt&nrm=iso>. Acesso em 25 jul. 2015. <http://dx.doi.org/10.1590/S1415-65552006000100004>.

GUTHRIE, Scott. et al. **Building Cloud Apps with Microsoft Azure**: Best practices for DevOps, data storage, high availability, and more. Redmond: Microsoft Press, 2014.

HP. **DevOps and OpsDev**: How Maturity Model Works. 2013. Disponível em <<http://h30499.www3.hp.com/t5/Business-Service-Management-BAC/DevOps-and-OpsDev-How-Maturity-Model-Works/ba-p/6042901>>. Acesso em 4 de jul. 2015.

HUMBLE, Jez; FARLEY, David. **Continuous Delivery**: Reliable Software Releases through Build, Test, and Deployment Automation. Boston: Addison-Wesley Professional, 2010.

HÜTTLERMANN, Michael. **DevOps for Developers** (Expert's Voice in Web Development). New York: Apress. 2012.

KIM, Gene. **Top 11 Things You Need to Know About DevOps**. 2012. Disponível em <<https://www.thinkhdi.com/~media/HDICorp/Files/White-Papers/whtppr-1112-devops-kim.pdf>>. Acesso em 04 jul. 2015.

KIM, Gene; BEHR, Kevin; SPAFFORD, George. **The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win.** [S.I.]: IT Revolution Press, 2013.

LORIERI, Leonardo. **DevOps com exemplos práticos.** [S.I.]: QCon, 2015. 1 vídeo de apresentação em evento; 1 slide. Disponível em <<http://www.infoq.com/br/presentations/devops-com-exemplos-praticos>>. Acesso em 15 de jul. 2015.

LOUKIDES, Mike. **What is DevOps?** Cambridge: O'Reilly Books. 2012.

LUCAS, Diogo. **Tratores e DevOps: colhendo os frutos da automação.** [S.I.]: QCon, 2015. 1 vídeo de apresentação em evento; 1 slide. Disponível em <<http://www.infoq.com/br/presentations/tratores-e-devops>>. Acesso em 18 de jul. 2015.

O'REILLY, Tim. **Operations: The New Secret Sauce.** O'Reilly Media. 2006. Disponível em <<http://radar.oreilly.com/2006/07/operations-the-new-secret-sauc.html>>. Acesso 04 jul. 2015.

PHILLIPS, Andrew. **No quick fix for DevOps.** 2013. Disponível em <<http://www.wired.com/2013/09/no-quick-fix-for-devops/>>. Acesso em 5 jul. 2015.

PUPPETLABS. **2014 State of DevOps Report.** 2014 Disponível em <<http://puppetlabs.com/sites/default/files/2014-state-of-devops-report.pdf>>. Acesso em 28 out. 2014.

ROBBINS, Jesse. **Operations is a competitive advantage...** (Secret Sauce for Startups!). O'Reilly Media. 2007. Disponível em <<http://radar.oreilly.com/2007/10/operations-is-a-competitive-ad.html>>. Acesso 04 jul. 2015.

SATO, Danilo. **DevOps na prática: entrega de software confiável e automatizada.** São Paulo: Caso do Código. 2014.

SILVA, Paulo Ricardo Ckless; RIBEIRO, José Luis Duarte. Uma proposta para a modelagem do valor percebido na prestação de serviços. **Prod.**, São Paulo , v. 12, n. 1, 2002 . Disponível em <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-65132002000100003&lng=pt&nrm=iso>. Acesso em 22 out. 2014.

THURMAN, Jonathan. **Unsung Tools of DevOps.** Cambridge: O'Reilly Books. 2013.

VASCONCELOS, Isabella Francisca Freitas Gouveia de. IBM: o desafio da mudança. **Rev. adm. empres.**, São Paulo , v. 33, n. 3, p. 84-97, June 1993 . Disponível em <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0034-75901993000300008&lng=en&nrm=iso>. Acesso em 25 Jul. 2015. <http://dx.doi.org/10.1590/S0034-75901993000300008>.

VIVEIROS, Daniel Vieira. **DevOps na nuvem com Google Cloud Platform e Docker.** [S.I.]: QCon, 2015. 1 vídeo de apresentação em evento; 1 slide. Disponível em

<<http://www.infoq.com/br/presentations/devops-na-nuvem-com-google-cloud-platform-docker>>. Acesso em 18 de jul. 2015.

WALLS, Mandi. **Building a DevOps culture**. Cambridge: O'Reilly Books. 2014.

WILLS, John. **The Convergence of DevOps**. IT Revolution Press. 2012. Disponível em <<http://itrevolution.com/the-convergence-of-devops/>>. Acesso em 04 jul 2015.

WOOD JR., Thomaz; CALDAS, Miguel P.. Antropofagia organizacional. **Rev. adm. empres.**, São Paulo , v. 38, n. 4, dez. 1998 . Disponível em <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0034-75901998000400002&lng=pt&nrm=iso>. Acesso em 22 out. 2014.

YIN, Robert K. **Estudo de caso: planejamento e métodos**. Porto Alegre: Bookman, 2001.

ZANELLI, José Carlos; SILVA, Narbal. **Interação humana e gestão: construção psicossocial das organizações de trabalho**. São Paulo: Casa do Psicólogo, 2012.

ZWIEBACK, Dave. **DevOps Hiring**. Cambridge: O'Reilly Books. 2014.