

Utilização de Microservices em Sistema de Gestão Empresarial - Estudo de Caso na Emissão de Nota Fiscal de Serviço Eletrônica

Deivid Dias Rodrigues(1); Walteno Martins Parreira Júnior(2)

(1) Pós-graduando em Análise e Desenvolvimento de Sistemas Aplicados a Gestão Empresarial, no Instituto Federal do Triângulo Mineiro, Campus Uberlândia Centro; Uberlândia, Minas Gerais; deivid.d.rodrigues@gmail.com; (2) Mestre, Professor do IFTM – Campus Uberlândia-Centro; waltenomartins@iftm.edu.br.

RESUMO: No Brasil as empresas prestadores de serviços, precisam declarar ao fisco que houve prestação de um serviço para que as prefeituras façam o recolhimento do ISSQN (Imposto Sobre Serviços de Qualquer Natureza). A forma de realizar essa prestação de contas é a emissão de um documento eletrônico que é enviado às prefeituras para que as mesma façam esse recolhimento. Nesse ponto entra os sistemas que auxiliam na gestão das empresas os ERP's que geralmente são sistemas baseados em uma arquitetura monolítica, e que fornecem recursos para para integração e envio desse documentos de forma automática por meio de web service disponibilizado pelas prefeituras onde cada uma tem o sua própria especificação, fazendo com que os sistemas tenham que lidar com essas especificidades de cada município para atender as várias prefeituras, nesse ponto existe um problema de implementação, disponibilidade e manutenção nos sistemas monolíticos. Diante disso, este trabalho tem como objetivo propor melhorias no processo de integração destes web services nos sistemas que foram projetados com uma arquitetura monolítica. Como resultado desta etapa do trabalho, foi realizado um estudo de caso descritivo por meio de análise de um ERP e de documentos que especificam a arquitetura de microservices, no sentido de propor uma solução para resolver o problema de implementação e disponibilidade apresentado acima, com o objetivo de diminuir o tempo de implementação e disponibilização utilizando a arquitetura de microservices, e também apresentando os ganhos e benefícios que podem ser alcançados utilizando microservices.

Palavras-chave: NFS-e. Microservice. ERP.

INTRODUÇÃO

Nas últimas décadas, muito tem se falado dos ERP's (Enterprise Resource Planning), que são sistemas de informação que integram todos os dados e processos de uma organização, a integração pode ser vista sob a perspectiva funcional como sistemas de finanças, contabilidade, recursos humanos, fabricação, marketing, vendas, compras, e sob a perspectiva sistêmica como, por exemplo, sistema de processamento de transações, sistemas de informações gerenciais, sistemas de apoio a decisão (LAUDON, PADOVEZE). Um sistema ERP tem como principal objetivo à integração de toda uma organização em seus diversos níveis funcionais através do armazenamento de informações colhidas em todos os setores empresariais. Sendo assim o sucesso de uma organização pode depender de um sistema ERP eficiente, o que pode causar um grande impacto nas estratégias corporativas da organização. A empresa, os clientes e qualquer indivíduo que interagir com o ERP pode ser beneficiado com as informações fornecidas pelo sistema.

A NFS-e (Nota Fiscal de Serviço Eletrônica) é um documento de natureza eletrônica gerada pelo contribuinte em seu estabelecimento, para declarar ao fisco que houve uma prestação de um serviço e conseqüentemente o recolhimento do ISSQN (Imposto Sobre Serviços de Qualquer Natureza). Esse documento é armazenado eletronicamente pela prefeitura ou entidade conveniada, que é responsável por desenvolver e manter um ambiente digital que seja capaz de receber, processar e gerar a NFS-e no prazo estipulado pela legislação municipal.

As prefeituras disponibilizam web services que geralmente são integrados nos ERP's para a transmissão automática dos documentos em formato XML para as prefeituras, porém como cada município tem sua própria solução torna complicado manter, integrar novos provedores aos sistemas e a inserção de novos colaboradores ao projeto visto que é necessário conhecimento avançado para que alterações não possam refletir de forma negativa inserindo erros em integrações que tecnicamente

estão funcionando, isso se dá pela estrutura monolítica do sistema onde as implementações para os diversos municípios ficam acopladas compartilhando recursos em comum entre eles e também contendo implementações de suas particularidades.

Em meio a muitas arquiteturas existentes, depara-se com um novo conceito de arquitetura o de microservices. Segundo Lewis e Fowler (2014) é uma abordagem para o desenvolvimento de um pedido único, como um conjunto de pequenos serviços, cada um executando em seu próprio processo e se comunicando com mecanismos leves, muitas vezes, uma API de recursos HTTP. Estes serviços são construídos em torno de capacidades de negócios e independentemente implementável por máquinas de implantação totalmente automatizada. Há um mínimo de gerenciamento centralizado destes serviços, que podem ser escritos em diferentes linguagens de programação e utilizam diferentes tecnologias de armazenamento de dados.

Essa pesquisa se propõem a apresentar como a arquitetura de microservices pode melhorar a manutenção e a implementação de novos municípios e também diminuir o prazo de entrega dessas implementações, levando em consideração todas as particularidades de cada município.

MATERIAL E MÉTODOS

Devido ao grande avanço da tecnologia nos últimos anos, a exigência dos usuários por performance, agilidade, usabilidade e também o crescente uso da computação na nuvem que possibilita o fácil acesso aos sistemas por diversos dispositivos e também disponível á todo tempo e em todo lugar, vem surgindo a necessidade de uma arquitetura de sistemas que atenda a essas necessidades de forma rápida e escalável.

No cenário atual do ERP, o processo de implementação e manutenção das notas de serviços eletrônicas é complicado e demanda tempo na disponibilização para o cliente final além de exigir alto conhecimento do processo pelo desenvolvedor, analisando os processos atuais pode ser observado que a arquitetura a qual se encontra não oferece facilidade de melhoria, sendo assim, buscando alternativas para resolver o problema nos deparamos com esse novo conceito de arquitetura de software denominado microservices que será usando como base para uma futura implementação do projeto.

Segundo Fowler (2014), para explicar a arquitetura Microservice é útil compará-la com o estilo monolítico, que é um aplicativo construído como uma única unidade, estes são muitas vezes construídos em três partes principais: interface, banco de dados e um aplicativo do lado do servidor. As aplicações monolíticas podem ser bem sucedidas, porém cada vez mais os desenvolvedores estão sentido frustrações com elas, especialmente á medida que mais aplicativos estão sendo implantados para a nuvem. Porque uma mudança feita para uma pequena parte do sistema, requer a reconstrução e implantação de todo o monólito este é o cenário do problema exposto acima que com o tempo torna difícil manter a modularização, e também escalar requer o dimensionamento de todo o aplicativo, em vez de partes que exigem mais recursos.

As dificuldades que a arquitetura monolítica apresenta na sua implantação, escalabilidade e implementação fez com que a surgisse a arquitetura de microservices, onde as aplicações são construídas como suítes de serviços, tornando esses serviços independentes de implementação, escalabilidade e implantação, além de poderem ser escritos em linguagens de programação diferentes e serem gerenciados por equipes diferentes.

A criação de componentes é uma das formas de se construir microservices, onde entende-se por componentes uma unidade do software que pode ser independente, substituível e atualizável, a principal forma de utilizar os microservices são dividi-los em serviços, uma das razões de utilizar serviços como componentes é que eles são independentemente implementáveis. Usar serviços como estes tem suas desvantagens, por exemplo, chamadas remotas são mais caras do que chamadas em processos, e assim as API remotas precisam ser mais complexas o que muitas vezes é mais complicado de usar.

A comunicação dos aplicativos criados a partir de microservices serão tão dissociado e coeso quanto possível, eles têm que possuir sua própria lógica de domínio e agir mais como filtros que recebem uma solicitação, aplicando a lógica adequada e produzindo uma resposta. Estes são coreografados utilizando protocolos RESTish simples, em vez de protocolos de comunicação complexos. Em um monólito, os componentes são executados em processo e a comunicação entre eles

é através de qualquer método de invocação ou chamada de função. O maior problema na mudança de um monólito para microservices reside na mudança do padrão de comunicação.

Bem como a descentralização das decisões sobre modelos conceituais, microservices também descentraliza decisões de armazenamento de dados. Enquanto as aplicações monolíticas preferem um único banco de dados lógico para dados persistentes, as empresas muitas vezes preferem um único banco de dados em uma variedade de aplicações. Microservices preferem deixar cada serviço gerir a sua própria base de dados, sejam diferentes instâncias da mesma tecnologia de banco de dados, ou inteiramente diferentes sistemas de banco de dados essa abordagem é chamada Persistência Poliglota.

Descentralizando a responsabilidade por dados através dos microservices tem implicações para gerenciar atualizações. A abordagem comum para lidar com atualizações é utilizar as transações para garantir a consistência ao atualizar vários recursos. Esta abordagem é muitas vezes usada dentro monólitos.

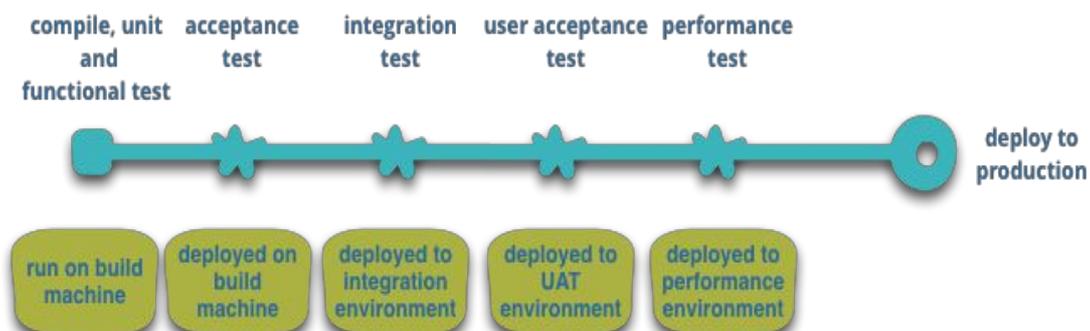
Usando transações contribui com consistência, mas impõe acoplamento temporais significativas, o que é problemático em vários serviços. Transações distribuídas são notoriamente difíceis de implementar e tem como consequência arquitetural dos Microservices enfatizar a coordenação entre os serviços transacionais, com reconhecimento explícito de que a consistência pode ser apenas eventual, os problemas devem ser tratados por meio de operações de compensação.

Gerenciar inconsistências desta forma é um novo desafio para muitas equipes de desenvolvimento, mas muitas vezes coincide com a prática de negócios. Muitas vezes as empresas lidam com um certo grau de incoerência, a fim de responder rapidamente à demanda, apesar de terem algum tipo de processo de reversão para lidar com erros.

A infraestrutura é outro ponto importante a ser analisado pois como os microservices trabalham com sistemas distribuídos, técnicas de automatização de infraestrutura devem ser levadas em consideração. Técnicas de automação de infraestrutura têm evoluído muito nos últimos anos como a evolução da nuvem com o AWS e, em particular, tem reduzido a complexidade operacional de construção, implantação e microservices operacionais.

Muitos dos produtos ou sistemas, construídos com microservices estão sendo construídos por equipes com vasta experiência de Distribuição Contínua e é precursor da Integração Contínua. As equipes que constroem software dessa forma fazem uso extensivo de técnicas de automação de infraestrutura. Isto é ilustrado na imagem de construção básica de microservices (Figura 1).

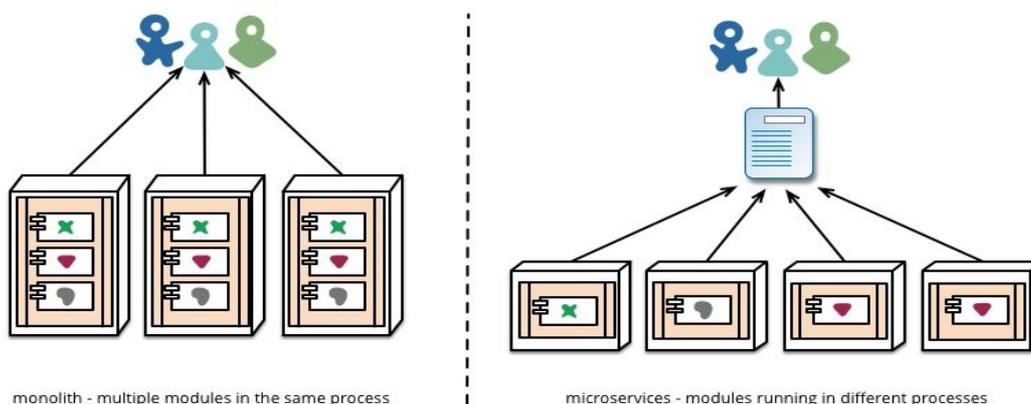
Figura 1 - Infraestrutura de construção básica de microservices.



Fonte: Fowler (2014)

Outra área onde a automação extensiva da infraestrutura é usada é na gestão dos microservices em produção. Em contraste com a afirmação acima que, enquanto implantação é chato, não há muita diferença entre monólitos e microservices, a paisagem operacional para cada um pode ser muito diferente. Conforme pode ser observado na Figura 2:

Figura 2 - Módulo de implantação.



Fonte: Fowler (2014)

Uma das consequências do uso de serviços como componentes, é que os aplicativos precisam ser projetados de modo que eles podem tolerar o fracasso dos serviços. Qualquer chamada de serviço pode falhar devido à indisponibilidade do fornecedor, o cliente tem que responder a este tão graciosamente quanto possível. Esta é uma desvantagem em comparação com um design monolítico em que introduz uma complexidade adicional para lidar com isso. A consequência é que as equipes que trabalham com microservice precisam refletir constantemente sobre como falhas de serviço que podem afetar a experiência do usuário.

Uma vez que os serviços podem falhar a qualquer momento, é importante as equipes serem capazes de detectar as falhas rapidamente e, se possível, automaticamente restaurar serviço. Aplicações Microservice colocam muita ênfase no monitoramento em tempo real do aplicativo, verificar os elementos arquitetônicos como quantas solicitações por segundo o banco de dados está recebendo e métricas relevantes de negócio como quantos pedidos por minuto são recebidas. Monitoramento da semântica pode fornecer um sistema de alerta quando algo dar errado e que aciona as equipes de desenvolvimento para acompanhar e investigar.

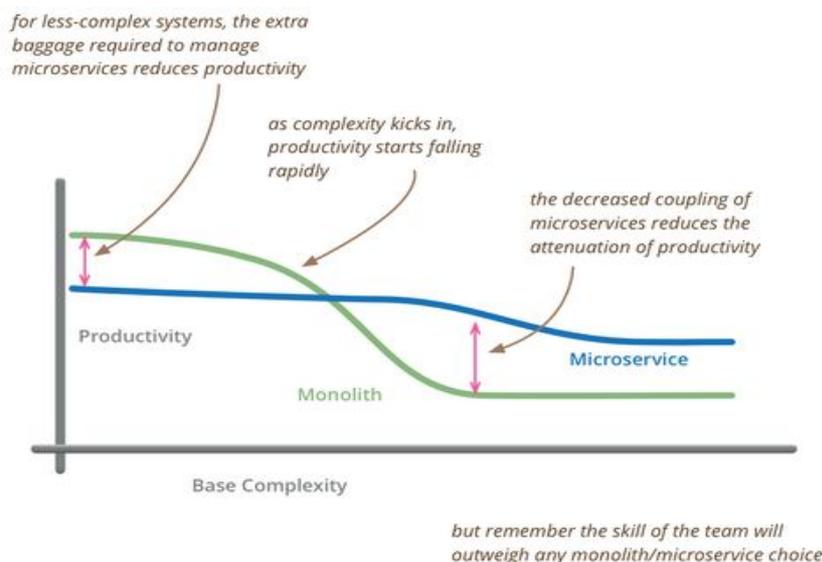
A arquitetura de microservices é uma ideia importante digna de consideração séria para aplicações empresariais. Porém existe as considerações que devem ser feitas, quanto a experiência da equipe, controle transacional e quando a recuperação de falhas o monitoramento desta é vital para detectar mau comportamento emergente rapidamente para que ele possa ser corrigido, evitando transtornos ao usuário final.

RESULTADOS E DISCUSSÃO

A utilização da arquitetura de microservices proporciona através de sua metodologia a melhoria no processo de implementação da nota fiscal de serviço eletrônico, onde os desenvolvedores ganharam tempo ao implementar um novo provedor, as empresas ganharam tempo e credibilidade pois irá implantar a modificação mais rápido para usuário final, o sistema será mais consistente com menor probabilidade de inserção de erros em processo existentes e também a equipe irá ganhar com a facilidade de inserção de novos membros ao projeto. E esta abordagem evitará os gargalos e dificuldades encontradas hoje na implementação das NFS-e, isso será possível com as técnicas já mencionadas da arquitetura e com uma equipe preparada e de alto nível para disponibilizar as modificações mais rápido evitando transtornos para o usuário final.

A Figura 3 representa o ganho de produtividade de acordo com o aumento da complexidade do sistema, pode-se observar que no início os sistemas monolíticos são mais produtivos por ser menores e menos complexos porém com o passar do tempo que novas regras e recursos vão sendo inseridos aumentando complexidade e consequentemente a produtividade diminui.

Figura 3 - Representação da produtividade em sistemas complexos.



Fonte: Fowler (2014)

Essa constante produtividade em microservices se dá pelo conjunto de trabalho e esforço que devem ser despendido para se usá-lo como implantação automatizada, monitoramento, saber lidar com o fracasso, eventual inconsistência e outros fatores que são introduzidos pela complexidade de se trabalhar com sistemas distribuídos.

A complexidade do sistema monolítico atual faz se necessário e se enquadra nas recomendações para utilização de microservices, depois da análise realizada para construção deste foi identificado como resultado que essa estrutura arquitetônica trará ganhos nos pontos críticos atuais de construção do sistema, aumento na produtividade consequentemente diminuindo o tempo de implementação das notas fiscais de serviço eletrônicas e na disponibilização destas implementações para o usuário final de forma transparente sem que o mesmo atualize todo o sistema para ter acesso a um novo recurso ou uma nova correção.

CONSIDERAÇÕES FINAIS

O presente trabalho tem como principal objetivo o estudo e avaliação do processo de implementação e manutenção de notas de serviços eletrônicas. Com o estudo realizado junto ao sistema de ERP, foi possível identificar o funcionamento e os procedimentos seguidos para implementar e manter essas NFS-e.

Através de pesquisas visando melhorar o processo, facilitando e agilizando-o, foi elaborado este documento que apresenta os benefícios de utilizar esse novo conceito de arquitetura de software, tal pesquisa será aprofundada para adequações ao atual sistema para implementação cujo objetivo visa melhorar a entrega final, facilidade de inserção de novos membros no projeto e diminuição no tempo de implementação de notas de serviços eletrônicas para novas prefeituras.

AGRADECIMENTOS

Gostaria de agradecer primeiramente a Deus, minha família, minha namorada que me apoia e me dá força para fazer o melhor e também a instituição IFTM (Campus Uberlândia Centro), que proporcionou um curso de qualidade ímpar, com corpo docente à altura, que corresponderam além das expectativas, fazendo jus ao nome da instituição. Agradecer também ao grande amigo Daniel Vieira que disponibilizou um tempo de sua atenção para avaliar e agregar seu conhecimento ao trabalho e ao Mestre, Professor; Walteno Martins Parreira Júnior que me orientou na construção desse trabalho enriquecendo o mesmo com seu conhecimento.

REFERÊNCIAS

- FOWLER, Martin **Microservice Premium**. 13 mai. 2015. Disponível em: <<http://martinfowler.com/bliki/MicroservicePremium.html>>, acesso em: 01 out. 2015.
- LAUDON, Kenneth C. **Sistemas de Informações gerenciais: administrando a empresa digital**. São Paulo: Prentice Hall, 2004. 61p.
- LEWIS James, FOWLER Martin. **Microservices**. 25 mar. 2014. Disponível em: <<http://martinfowler.com/articles/microservices.html#ComponentizationViaServices>>. Acesso em 21 set. 2015.
- NEWMAN, Sam. **Building Microservices**. Sebastopol: O'Reilly, 2015.
- NAMIOT Dmitry; SNEPPE Manfred. **On Micro-services Architecture**. International Journal of Open Information Technologies, n. 2, v.9, 2014.
- PADOVOZE, Clóvis Luís. **Sistemas de informações contábeis: fundamentos e análise**. São Paulo: Atlas, 2004. 68p.
- RICHARDSON, Chis. **Microservices: Decomposição de Aplicações para Implantação e Escalabilidade**. Disponível em: <<http://www.infoq.com/br/articles/microservices-intro>>, acesso em 21 set. 2015.
- UHLE, Johan. **On Dependability Modeling in a Deployed Microservice Architecture**. Hasso Plattner Institut, Universität Potsdam, 2014.

Referencias:

RODRIGUES, Deivid Dias; PARREIRA JUNIOR, Walteno Martins. Utilização de Microservices em Sistema de Gestão Empresarial Estudo de Caso na Emissão de Nota Fiscal de Serviço Eletrônica. In: Simpósio de Pós-Graduação do IFTM (Simpós), 2., 2015. **Anais...** Uberaba: IFTM, 2015. ISSN 2359-0130.