

Plano de Trabalho de Conclusão de Curso

Análise e caracterização de arquiteturas de microserviços empregados a jogos MMORPG voltada a otimização do uso de recursos de gerenciamento de mundos virtuais

Marlon Henry Schweigert – marlon.henry@magrethealabs.com
Charles Christian Miers – charles.miers@udesc.br (*orientador*)

Turma 2018/1 – Joinville/SC

1 de Fevereiro de 2018

Resumo

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. **Palavras-chave:** *arquitetura de microserviços, desenvolvimento de jogos, rede de jogos, otimização de recursos*

1 Introdução e Justificativa

Os avanços tecnológicos de sistemas distribuídos estão permitindo que pessoas utilizem de serviços com volumes massivos de dados para aplicações sensíveis a latência. Essa situação é bem favorável a área de jogos massivos, tendo atraído pesquisadores para testar e validar novas abordagens com o objetivo de reduzir a carga e o impacto a latência para o usuário final nesses serviços, resultando em uma melhor experiência aos jogadores da categoria de jogo tratado no presente trabalho[Kim, Kim e Park 2008].

O mercado de jogos massivos multijogadores de interpretação (*MMORPG*) vem crescendo desde 2012 [Bilton 2011], sendo no ano de 2016 um dos mais lucrativos[Statista 2016]. A sua projeção para 2018 é que sejam arrecadados mais de 30 bilhões de dólares americanos com esta categoria de jogos [Statista 2017], um aumento de 20% a mais sobre o ano de 2016.

Massively Multiplayer Online ou *MMO* (como são popularmente conhecidos) são os jogos de interpretação multijogador massivos. A principal característica desse estilo de jogo é a comunicação e representação virtual de um mundo fantasia onde cada jogador pode interagir com objetos virtuais compartilhados ou tomar ações sobre outros de jogadores em tempo real, tendo como principais objetivos a resolução de problemas conforme a sua regra de design, o desenvolvimento do personagem e a interação entre os jogadores[Hanna 2015].

Serviços MMO são utilizados como negócio viável e lucrativo, a qual a experiência de uso do usuário final é um fator crítico para o sucesso. A maioria dos jogos massivos disponíveis no mercado estão implementados sobre uma arquitetura com diversos servidores, onde o desempenho destes servidores influenciam na experiência do usuário final. Modelar um sistema de alto desempenho em torna-se um trabalho essencial para a satisfação do usuário final[Huang, Ye e Cheng 2004].

1.1 Ocorrências em serviços massivos

Uma métrica popular para mensurar o desempenho do jogo é o número de conexões. Em geral, se o servidor caso ultrapasse esses limites, diversas ocorrências de experiência com o usuário poderão ocorrer. As ocorrências comuns são[Huang, Ye e Cheng 2004]:

- **Longo tempo de resposta aos clientes:** esta ocorrência trás uma qualidade insatisfatória ao usuário ou até mesmo impossibilitando o uso do serviço.
- **Dessincronização com os clientes:** esta ocorrência promove reversão na aplicação, proporcionando desconforto e baixa fluidez do jogo.
- **Problemas internos ao servidor:** esta ocorrência pode estar ligada a diversos outros erros como sobrecarga no banco de dados, gerenciamento lento do espaço ou inconsistências dentro do jogo perante a regra de negócios.
- **Falha de conexão entre o cliente e os micros serviços:** Promove a inacessibilidade do serviço ao usuário final.

Existem algumas causas comuns para essas falhas, sendo alguns dos problemas conhecidos que podem causar tais ocorrências[Huang, Ye e Cheng 2004]:

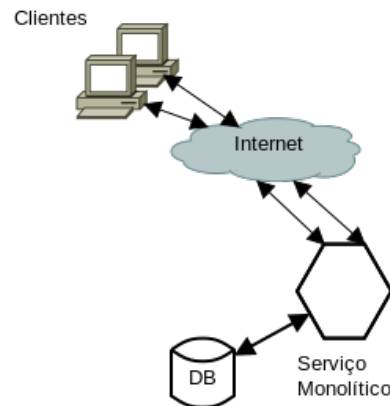
- **Baixo poder computacional do servidor:** poder computacional baixo para a qualidade de experiência do usuário final desejada.
- **Complexidade de algoritmos:** o serviço usa algoritmos de alta complexidade ou até mesmo a regra de negócios que demandem de algoritmos complexos.
- **Limitado pela própria arquitetura:** está limitado a arquitetura que compõe o serviço.
- **Uma arquitetura que não foi planejada ou testada para receber determinada carga:** está limitado diretamente pelo número de conexões.

1.2 Arquiteturas de micros serviços

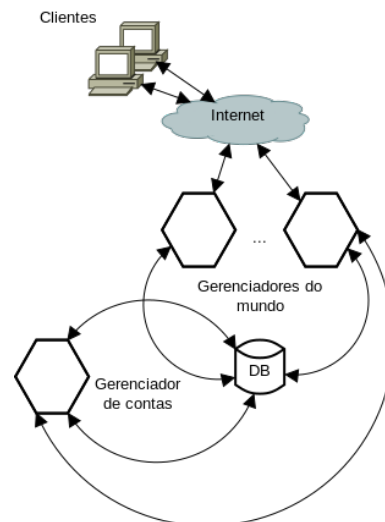
As ocorrências descritas são resolvidas no segmento de desenvolvimento web utilizando uma arquitetura de micros serviços. Entende-se por micros serviço as aplicações que executam operações menores de um macroserviço[Willson 2017]. Essas arquiteturas iniciam uma nova linha de desenvolvimento de aplicações preparadas para executar sobre nuvens computacionais, promovendo maior flexibilidade,

escalabilidade, gerenciamento e desempenho, sendo a principal escolha de arquitetura de grandes empresas como Amazon, Netflix e LinkedIn[Khazaei et al. 2016][Villamizar et al. 2016].

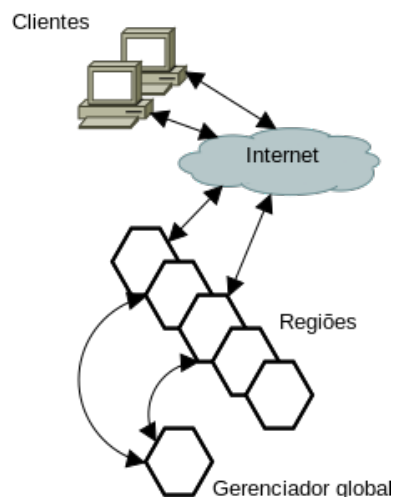
A arquitetura ingênua, ou monolítica, é dada por uma única instância agregada com um único banco de dados (relação 1-1). Essa implementação foca somente em criar uma interface global do jogo através desse servidor, recebendo as ações do jogador e retornando o estado de jogo para o cliente.



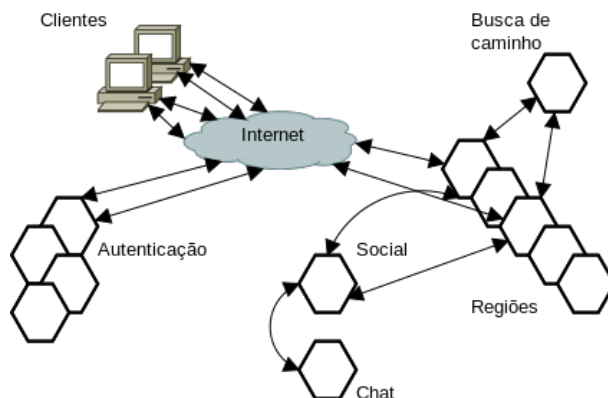
A arquitetura de múltiplos canais[Huang, Ye e Cheng 2004] é uma arquitetura de microserviços que baseia-se na divisão dos clientes em instâncias de jogos diferentes, utilizando a mesma base de dados (relação n-1). Se faz necessário um gerenciador de contas, para impedir multiplas conexões em canais diferentes da mesma conta.



A arquitetura baseada em regiões[Salz 2016] basea-se na distribuição uniforme dos jogadores no mundo virtual, visando distribuir os jogadores dentre os múltiplos serviços por região. Os bancos de dados foram retirados para facilitar a visualização.



A arquitetura baseada em serviços[Willson 2017] aborda o jogo como um único servidor monolítico, separando as suas funcionalidades em microserviços. Esta arquitetura arrisca tempo de resposta para obter o máximo de conexões no macroserviço. Também utiliza a abordagem de separação das regiões, porém utiliza divisão dinâmica do mundo.



1.3 Justificativa

A proposta de otimização das análises realizadas sobre as arquiteturas de microserviços para jogos massivos focada ao gerenciamento de mundos virtuais proposta pela literatura traz impacto direto a melhoria da qualidade de experiência ao usuário final, por sua vez, proporcionando soluções com melhor garantia de sucesso sobre outras arquiteturas problemáticas.

2 Objetivos

Análise e caracterização de arquiteturas de microserviços empregados a jogos MMORPG voltada a otimização do uso de recursos (*e.g.*, *banda, memória, processamento*[Huang, Ye e Cheng 2004]) de gerenciamento de mundos virtuais. Propor soluções tangíveis (*e.g.*, *protocolos, compressão, paralelismo*[Huang, Ye e Cheng 2004]) para otimizar os recursos utilizados pelas arquiteturas analisadas.

Os objetivos específicos são:

- Identificar e caracterizar arquiteturas empregadas na categoria de jogos do presente trabalho.
- Identificar e caracterizar os protocolos utilizados e disposição dos componentes nessas arquiteturas.
- Identificar e analisar ferramentas de análise de recursos para definir métricas as arquiteturas identificadas e caracterizadas.
- Aplicar as arquiteturas descritas na literatura em uma nuvem de computadores *OpenStack*.
- Analisar o comportamento das arquiteturas aplicadas, levantando questões de desempenho e recursos utilizados.
- Propor alternativas de otimização para os problemas encontrados nas devidas arquiteturas.

3 Metodologia

Para que seja possível atingir os objetivos, serão utilizados dois métodos: pesquisa referenciada, desenvolvida durante o Trabalho de Conclusão de Curso I, e pesquisa aplicada, desenvolvida durante o Trabalho de Conclusão de Curso II.

Na pesquisa referenciada serão levantadas arquiteturas da literatura, buscando as mais adequadas ao escopo deste trabalho. Será dividido em três situações: levantamento de arquiteturas de micro-serviços descritas na literatura e caracterização dessas arquiteturas. Por fim, o levantamento de possíveis simulações para efetuar testes durante a pesquisa aplicada.

Na pesquisa aplicada, o resultado a ser obtido é a análise das arquiteturas de microserviços caracterizadas, visando uma profunda análise sobre os recursos computacionais consumidos e identificação de seus gargalos. Divide-se em três situações: aplicação das arquiteturas descritas e selecionadas, realização dos testes utilizando a simulação descrita na pesquisa referenciada e análise dos dados coletados.

Para que os resultados sejam alcançados, são definidas as seguintes etapas:

1. **Levantamento e fichamento das referências:** Pesquisa de fontes para embasamento teórico do trabalho, com base nos objetivos específicos;
2. **Consolidação das referências:** Compreensão e seleção de artefatos literários que permitam atingir o objetivo do Trabalho de Conclusão de Curso I;
3. **Identificação e caracterização de arquiteturas descritas na literatura:** Enumeração e caracterização das arquiteturas de microserviços descritas, bem como se elas empenham em solucionar problemas de gerenciamento espacial;
4. **Identificação e caracterização de simulações aplicáveis ao teste:** Eleger e caracterizar a simulação a ser aplicada nos testes que demandem das arquiteturas;

5. **Especificação das arquiteturas selecionadas:** Especificar o funcionamento das arquiteturas selecionadas.
6. **Escrita Trabalho de Conclusão de Curso I;**
7. **Aplicação das arquiteturas selecionadas na pesquisa referenciada:** Desenvolvimento das arquiteturas descritas para aplicação dos testes;
8. **Aplicação da simulação descrita na pesquisa referenciada:** Desenvolvimento da simulação para interagir com as arquiteturas de microserviços;
9. **Realização dos testes utilizando a simulação descrita na pesquisa referenciada:** Aplicação em uma nuvem de computadores metrificando os dados obtidos;
10. **Análise das arquiteturas testadas:** Analisar as métricas obtidas e descrever resultados, identificando possíveis gargalos nas arquiteturas;
11. **Propor otimizações para melhorar as métricas obtidas:** Analisar pontos de gargalo nos microserviços analisados e propor soluções viáveis para aumentar o desempenho desses sistemas.
12. **Escrita Trabalho de Conclusão de Curso II;**

4 Cronograma proposto

Etapas	2018											
	J	F	M	A	M	J	J	A	S	O	N	D
1												
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												

5 Linha de Pesquisa

Este trabalho será desenvolvido junto ao Grupo de Redes e Aplicações Distribuídas (GRADIS) e ao Laboratório de Processamento Paralelo e Distribuído (LabP2D). Esta pesquisa abrange as áreas de Redes de Computadores, Sistemas Distribuídos, Segurança em Redes de Computadores, Processamento Paralelo e Engenharia de Software.

6 Forma de Acompanhamento/Orientação

O acompanhamento será realizado principalmente através de reuniões semanais ou quinzenais, com duração máxima de 1 (uma) hora. Eventualmente as reuniões poderam ser trocadas por vídeo-conferência, troca de mensagens de correio eletrônico ou telefone. O controle das tarefas a fazer serão feitas baseadas em uma ata gerada a cada reunião. Metas semanais ou quinzenais serão atribuídas para melhor acompanhamento.

Referências

- [Bilton 2011]BILTON, N. *Search Bits SEARCH Video Game Industry Continues Major Growth, Gartner Says*. 2011. Acessado em: 19/01/2018. Disponível em: <<https://bits.blogs.nytimes.com/2011/07/05/video-game-industry-continues-major-growth-gartner-says/>>.
- [Hanna 2015]HANNA, P. *Video Game Technologies*. 2015. Acessado em: 19/01/2018. Disponível em: <<https://www.di.ubi.pt/agomes/tjv/teoricas/01-genres.pdf>>.
- [Huang, Ye e Cheng 2004]HUANG, G.; YE, M.; CHENG, L. Modeling system performance in mmorpg. In: *IEEE Global Telecommunications Conference Workshops, 2004. GlobeCom Workshops 2004*. [S.l.: s.n.], 2004. p. 512–518.
- [Khazaei et al. 2016]KHAZAEI, H. et al. Efficiency analysis of provisioning microservices. In: *2016 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. [S.l.: s.n.], 2016. p. 261–268.
- [Kim, Kim e Park 2008]KIM, J. Y.; KIM, J. R.; PARK, C. J. Methodology for verifying the load limit point and bottle-neck of a game server using the large scale virtual clients. In: *2008 10th International Conference on Advanced Communication Technology*. [S.l.: s.n.], 2008. v. 1, p. 382–386. ISSN 1738-9445.
- [Salz 2016]SALZ, D. *Albion Online - A Cross-Platform MMO (Unite Europe 2016, Amsterdam)*. 2016. Disponível em: <<https://www.slideshare.net/davidsalz54/albion-online-a-crossplatform-mmo-unite-europe-2016-amsterdam>>.
- [Statista 2016]STATISTA. *Statistics and Facts on MMO/MMORPG gaming*. 2016. Acessado em: 19/01/2018. Disponível em: <<https://www.statista.com/topics/2290/mmo-gaming/>>.
- [Statista 2017]STATISTA. *Games market revenue worldwide in 2015, 2016 and 2018, by segment and screen (in billion U.S. dollars)*. 2017. Acessado em: 19/01/2018. Disponível em: <<https://www.statista.com/statistics/278181/video-games-revenue-worldwide-from-2012-to-2015-by-source/>>.
- [Villamizar et al. 2016]VILLAMIZAR, M. et al. Infrastructure cost comparison of running web applications in the cloud using aws lambda and monolithic and microservice architectures. In: *2016*

16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid). [S.l.: s.n.], 2016. p. 179–182.

[Willson 2017]WILLSON, S. C. *Guild Wars Microservices and 24/7 Uptime*. 2017. Disponível em: <[http://twvideo01.ubm-us.net/o1/vault/gdc2017/Presentations/Clarke-Willson_Guild Wars 2 microservices.pdf](http://twvideo01.ubm-us.net/o1/vault/gdc2017/Presentations/Clarke-Willson_Guild_Wars_2_microservices.pdf)>.

Charles Christian Miers

Marlon Henry Schweigert

Rafael Rodrigues Obelheiro
(Coordenador do GRADIS)