UNIVERSIDADE FEDERAL DE PERNAMBUCO
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
CENTRO DE INFORMÁTICA
2017.1

Utilização de técnicas de Process Mining em Sistemas de Middleware Adaptativos

Proposta de Trabalho de Graduação

Aluna: Rayana Vasconcelos de Sá Alencar (rvsa2@cin.ufpe.br)
Orientador: Nelson Souto Rosa (nsr@cin.ufpe.br)

Recife, 26 de Março de 2017

Sumário

1.	Contextualização	3
2.	Objetivos	5
3.	Cronograma	6
4.	Referências Bibliográficas	7
5.	Possíveis Avaliadores	8
6.	Assinaturas	9

Contextualização

Middleware é uma infraestrutura de software que se encontra entre a aplicação e o Sistema Operacional, redes e hardwares adjacentes [1]. O Middleware torna mais fácil o desenvolvimento de sistemas distribuídos, pois com ele os desenvolvedores de aplicações não precisam ter uma preocupação em relação ao dispositivos e a condição que o usuário vai utilizar a aplicação .

Imagine que um usuário está utilizando uma aplicação que você construiu. É esperado que durante a execução ocorram eventos que afetem o seu funcionamento, como a diminuição da bateria ou uma mudança de rede wi-fi no dispositivo. Portanto, a existência desses cenários torna necessário que existam também adaptações para diminuir o impacto desses eventos no funcionamento da aplicação em geral [3].

Como foi explicado, o middleware é o responsável por abstrair as configurações em que o dispositivo se encontra, portanto ele também deve ser capaz de solucionar problemas relacionado ao ambiente. Para isso, o middleware precisa ter a capacidade de se adaptar.

Um middleware adaptativo é um tipo de middleware que modifica o seu comportamento durante o tempo de execução, sem precisar parar o seu funcionamento [2]. É muito importante a existência de middleware adaptativos, especialmente para plataformas de IoT, pois envolvem diferentes tipos de dispositivos.

Em 2011 a IBM propôs uma nova arquitetura de software conhecida como MAPE-K (Monitoring, Analysis, Planning, Execution and Knowledge). Essa arquitetura foi desenvolvida para diminuir as dificuldades presentes na monitoração, principalmente em tempo de execução de sistemas, que a cada ano se tornavam mais heterogêneos e distribuídos [4]. A solução encontrada foi chamada de *autonomic computing*, são sistemas que conseguem automonitorar seu comportamento e que implementam a arquitetura MAPE-K. A arquitetura MAPE-K é composta de 4 módulos: Monitoramento, Análise , Planejamento e Execução.

O módulo Monitoramento, como o próprio nome sugere, é responsável por coletar a informações de como o sistema está funcionando. Em seguida, no módulo de Análise é feita uma investigação dessas informações para verificar se o comportamento do sistema está de acordo com as políticas previamente definidas pelo usuário. Se em algum momento, o analisador encontrou alguma violação de política, isso indica que o sistema não está funcionando como deveria. Então, o módulo Planejamento define quais modificações devem ser feitas e no módulo de Execução essas mudanças serão realizadas. Esse ciclo se repete durante todo o tempo de vida do software [4].

Para construir um Middleware adaptativo é preciso responder 3 principais perguntas: **Como** será feita a adaptação, **quando** ela deve ser realizada e **onde** o middleware precisará ser modificado [2]. Uma solução é implementar o middleware seguindo a arquitetura MAPE-K. O módulo de monitoramento vai gerar informações necessárias para o módulo de Análise que irá detectar problemas que porventura ocorram, além da fonte dos mesmos. Além disso, o usuário deve indicar quais são as políticas de reconfigurações do middleware para o sistema saber como se adaptar.

Uma área de pesquisa que vem crescendo cada vez mais é o de mineração de processos, que consiste em extrair informação a partir de *logs* dos sistemas. O objetivo da mineração é descobrir, monitorar e melhorar processos reais [5]. Assim, podemos utilizar a mineração de processos para analisar o comportamento de um middleware.

Atualmente existem diversas ferramentas de mineração de processos. Com o objetivo de conectar todas essas ferramentas, foi desenvolvido um framework chamado ProM[1], através do qual é possível utilizar diferentes modelos de mineração, desde Petri net até uso de fórmulas da linguagem temporal linear (LTL)[6].

[1] http://www.promtools.org/doku.php

Objetivos

O objetivo do trabalho é implementar a arquitetura MAPE-K em um middleware adaptável. Durante o módulo de análise da arquitetura, os logs coletados do middleware vão passar pelo *LTLMi*ner [7] uma ferramenta que invoca o *LTLChecker*, um plugin do ProM, para analisar o log de acordo com uma ou várias fórmulas LTL. O retorno dessa análise é que vai determinar se uma adaptação vai ser necessária ou não.

O projeto vai ser implementado em Java e no final vão ser realizados testes para medir o *overhead* de se utilizar essa técnica para implementação de um middleware adaptativo.

Cronograma

Atividades	Março	Abril	Maio	Junho	Julho
Elaboração da proposta					
Revisão da Literatura					
Implementação da ferramenta					
Escrita do trabalho					
Preparação da apresentação					

Referências Bibliográficas

[1] Schantz, R. E. and Schmidt, D. C. 2007. Middleware for Distributed Systems. Wiley Encyclopedia of Computer Science and Engineering.

[2] N. S. ROSA, “Middleware adaptation through process mining,” tech. rep., Universidade Federal de Pernambuco, 2016.

[3] P. McKinley, S. Sadjadi, E. Kasten, and B. Cheng, “Composing adaptive software,” Computer, vol. 37, no. 7, pp. 56–64,2004.

[4] J. O. Kephart, D. M. Chess, "The vision of autonomic computing", Computer, vol. 36, no. 1, pp. 41-50, Jan. 2003.

[5] W.M.P. van der Aalst, A. Adriansyah, "Process Mining Manifesto", Springer-Verlag Business Process Management Workshops, vol. 99, 2011.

[6] B. F. van Dongen, A. K. A. de Medeiros, H. M. W. Verbeek, A. J. M. M. Weijters, and W. M. P. van der Aalst, “The Prom Framework: A New Era in Process Mining Tool Support,” in Proceedings of the 26th International Conference on Applications and Theory of Petri Nets, ser. ICATPN’05. Berlin,Heidelberg: Springer-Verlag, 2005, pp. 444–454.

[7] Kasekamp, Tõnis. "Discovering LTL-Based Business Rules From Event Logs". Bachelor’s Thesis. UNIVERSITY OF TARTU, 2015. Print.

Possíveis Avaliadores

Possíveis avaliadores deste trabalho são os professores Carlos André Guimarães Ferraz e Kiev Santos da Gama.

Assinaturas

Rayana Vasconcelos de Sá Alencar

Orientando

Nelson Souto Rosa.

Orientador