

NOA: Um *Middleware* para Maximização das Perspectivas de Orquestração do Controlador SDN OpenDaylight

Kevin B. Costa¹, Emídio P. Neto¹, Charles H. F. dos Santos¹
Felipe S. Dantas Silva^{1,2}, Marcílio O. O. Lemos², Augusto Venâncio Neto²

¹Laboratório de Tecnologias Avançadas em Redes de Computadores (LaTARC)
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte (IFRN)
Natal-RN, Brasil

²Departamento de Informática e Matemática Aplicada (DIMAp)
Universidade Federal do Rio Grande do Norte (UFRN) – Natal-RN, Brasil

{kevin.barros, emidio.neto}@academico.ifrn.edu.br
{charles.hallan, felipe.dantas}@ifrn.edu.br
marcilio.cc.lemos@gmail.com, augusto@dimap.ufrn.br

Abstract. *In the context of the Software-Defined Networking (SDN) approach, SDN controllers are expected to be able to provide solutions that provide infrastructure management in a practical and efficient way. Based on this, this paper presents the Network Orchestration Agent (NOA), a middleware for the OpenDaylight controller which provides a set of functionalities to improve, through dynamically defined filtering rules, the amount of information that is made available to the applications, significantly reducing the number of event messages for external management entities, such as orchestrators. The results of the assessments, conducted under scalability perspectives in arbitrary topologies sizes, demonstrated the feasibility and effectiveness of NOA over the original OpenDaylight implementation.*

Resumo. *No contexto das Redes Definidas por Software (Software-Defined Networking – SDN), é esperado que os controladores sejam capazes de fornecer soluções que proporcionem um gerenciamento da infraestrutura de forma prática e eficiente. Com base nisso, este trabalho apresenta o Network Orchestration Agent (NOA), um middleware para o controlador OpenDaylight que provê um conjunto de funcionalidades para otimizar, por meio de regras de filtragem definidas dinamicamente, a quantidade de informações que são disponibilizadas para as aplicações, reduzindo significativamente a quantidade de mensagens de eventos para as entidades de gerenciamento externo, como orquestradores. Os resultados das avaliações, conduzidos sob perspectivas de escalabilidade em topologias de tamanhos arbitrários, demonstraram a viabilidade e eficácia do NOA frente ao OpenDaylight original.*

1. Introdução

O surgimento do paradigma de Redes Definidas por Software (*Software Defined Networking* – SDN) [Boucadairm and Jacquenet 2014] possibilitou, por meio da centralização das decisões de rede, a implantação de soluções inovadoras para atender às novas demandas globais da Internet. Com isso, a adoção desse paradigma em larga escala, por meio

de técnicas de migração adotadas nas infraestruturas de várias organizações [ONF 2014], exige dos controladores SDN a implementação de estratégias que forneçam ao operador de rede o controle ubíquo sobre os elementos da infraestrutura, bem como prover os melhores níveis de desempenho ao sistema durante a coleta de dados e aplicação de decisões de controle.

Dentre os principais controladores SDN atualmente disponíveis (e.g. *Floodlight*¹, *Ryu*², *ONOS*³ etc.) o OpenDaylight (ODL) [Medved et al. 2014] destaca-se ao apresentar funcionalidades para o monitoramento da infraestrutura da rede, como estruturas avançadas de dados providas por *DataStores* (i.e. um local conceitual para armazenamento, gerenciamento e distribuição de informação) [Enns et al. 2011] e interfaces baseadas em REST APIs [Fielding and Taylor 2000], que são utilizadas por entidades remotas de gerenciamento, como orquestradores de rede [Gomes et al. 2016], por exemplo. Tais funcionalidades permitem proatividade em diferentes aspectos no gerenciamento da infraestrutura, como diagnóstico de erros, avaliação de desempenho e identificação de padrões de tráfego.

Embora o OpenDaylight possua vantagens em relação ao fornecimento de uma variedade de informações sobre o estado da rede [Medved et al. 2014], a definição de sua arquitetura apresenta uma série de restrições no que diz respeito à definição das interfaces de comunicação internas entre as aplicações do controlador, cuja natureza afeta a praticidade no fornecimento das informações devido a grande quantidade de dados que são armazenados nos *DataStores*, utilizadas para acesso externo por meio da REST API. Além disso, o fornecimento dessas informações requer que o administrador da rede possua um conhecimento aprofundado sobre as funções do controlador e das especificidades de suas mensagens.

Nesse contexto, se faz necessário a implementação de métodos que possibilitem a sintetização das informações sobre o estado da rede para o uso de aplicações externas, de modo a possibilitar a aplicação de filtros que selecionem quais dados devem ser notificados durante os eventos da topologia, por exemplo. Esses mecanismos devem possibilitar que os dados sejam apresentados com o mínimo de complexidade para os operadores de rede, facilitando a leitura e aplicação das informações em sistemas de orquestração de infraestruturas.

Neste contexto, este trabalho apresenta o *Network Orchestration Agent* (NOA), um *middleware* para o controlador OpenDaylight, atualmente a maior instância⁴ entre os controladores SDN, que tem como objetivo principal maximizar o seu potencial de orquestração através de funcionalidades que são capazes de otimizar a quantidade de informações sobre os eventos da topologia de rede, assim como os fluxos de dados entre os elementos de interconexão em um ambiente SDN, disponibilizadas por meio de notificações em tempo real.

Os resultados das avaliações, conduzidas na perspectiva de destacar as melhorias providas pelo NOA, revelaram a eficácia da proposta frente ao OpenDaylight original em

¹<http://www.projectfloodlight.org/floodlight/>

²<https://osrg.github.io/ryu>

³<https://onosproject.org/>

⁴<https://www.opendaylight.org/use-cases-and-users>

termos de redução de mensagens de sinalização necessárias para a execução de procedimentos básicos de rede, tais como inicialização da topologia (*bootstrap*), descoberta de caminhos e desligamento (*tear down*).

O restante deste artigo está estruturado da seguinte forma: a Seção 2 fornece uma descrição detalhada da proposta, incluindo sua arquitetura, seus respectivos componentes e operações. A Seção 3 detalha a avaliação do NOA através de 4 (quatro) conjuntos de avaliação. Por fim, a Seção 4 fornece as considerações finais e delinea apontamentos para trabalhos futuros.

2. Network Orchestration Agent – NOA

O *Network Orchestration Agent* (NOA) consiste em uma proposta de extensão para o controlador OpenDaylight cujo objetivo é prover a otimização dos recursos de comunicação em tempo real da plataforma para fornecer um maior nível de ubiquidade de orquestração. Desta maneira, o NOA atua como um *middleware* que fornece um alto grau de abstração para a implantação de funcionalidades inovadoras (e.g. QoS, gerenciamento de mobilidade etc.) cujos requisitos de operacionalização necessitam de informações de eventos da topologia e serviços em tempo de execução. A Figura 1 fornece uma visão geral da proposta do NOA sob a arquitetura padrão do controlador OpenDaylight.

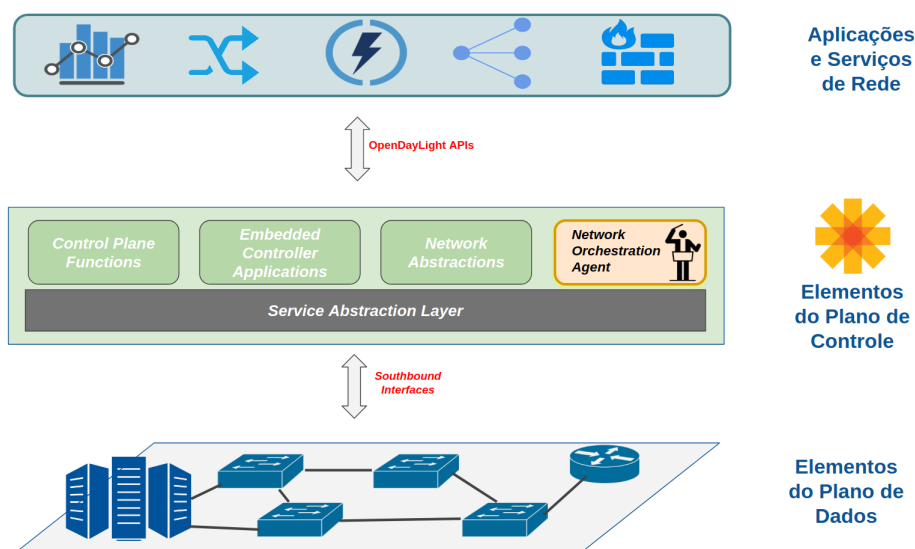


Figura 1. Visão geral da proposta NOA

Para ser capaz de realizar as operações propostas, o NOA foi projetado como um *middleware* posicionado sob a REST API do OpenDaylight de modo que possa se beneficiar da praticidade fornecida pelas interfaces do controlador para obtenção de informações dos *DataStores* responsáveis pela topologia e detalhes dos dispositivos (i.e. *Network Topology* e *OpenDayLight-Inventory:Nodes*) de maneira eficiente.

A Figura 2 apresenta a arquitetura modular do NOA, enfatizando, em seus componentes principais, as funções de controle, protocolos e interfaces. As subseções seguintes detalham os componentes principais da arquitetura proposta.

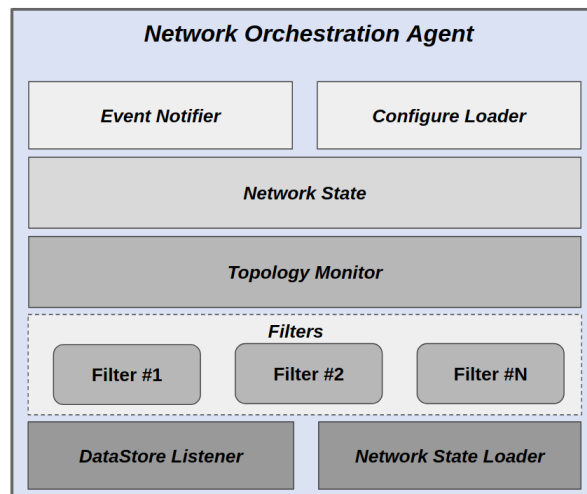


Figura 2. Arquitetura da proposta NOA

2.1. DataStore Listener

O *DataStore Listener* é responsável pela aquisição das atualizações do *DataStore* que armazena o estado da rede (i.e. *Network-Topology*), através do fluxo de dados que o notifica sobre as modificações (e.g. criação/remoção de *links*) em sua estrutura. É constituído de um gerador de fluxos, que é responsável por automatizar o processo de criação do endereço *websocket* a ser utilizado para receber as notificações sobre modificações, em tempo real, entre o NOA e o *DataStore Network-Topology*.

2.2. Network State Loader

O *Network State Loader* é a primeira entidade a ser executada quando o NOA é inicializado. Este componente tem como principal função carregar parâmetros estáticos (e.g. dados sobre topologias já conectadas) da infraestrutura para serem repassados ao sistema de filtragem do NOA, onde serão submetidas a um processo de classificação.

2.3. Filters

O componente *Filters*, elemento central da arquitetura NOA, foi projetado com o objetivo de fornecer ao operador de rede uma maior granularidade sobre a gerência dos eventos da infraestrutura. Por meio de uma abordagem baseada em técnicas de reconfiguração dinâmica de componentes é possível criar regras de filtragem (através de arquivos de configuração) baseadas em aspectos como: (i) categoria de elemento de rede (e.g. *hosts*, *switches*, *links* etc.); (i) eventos de *up/down* relacionados aos elementos de rede, tabelas de fluxos dos *switches*, dentre outros.

Mediante os recursos providos por este componente, NOA introduz a perspectiva de aprimoramento de novas soluções de gerenciamento de redes (e.g. QoS, mobilidade, segurança etc.) através da otimização do fluxo de mensagens de eventos relacionados aos interesses de cada novo componente a ser desenvolvido. Neste contexto, é possível aperfeiçoar o desempenho do controlador sob diversos panoramas, tais como tamanho e quantidade de mensagens por evento, sinalização etc.

2.4. Topology Monitor

O *Topology Monitor* analisa o resultado do processamento realizado pelo *Filters* e, baseado em uma tabela de estados da topologia (previamente construída a partir das informações do *Network State Loader* ou do *DataStore Listener*), decide se irá proceder com a atualização dos dados da tabela, bem como se essas atualizações serão encaminhadas às aplicações (i.e. para as aplicações que estiverem fazendo uso deste recurso).

2.5. Event Notifier

O *Event Notifier* é responsável por receber atualizações do *Topology Monitor* e atua como um encaminhador das mensagens para as entidades interessadas. Para fazer uso das funcionalidades deste componente, as entidades devem se inscrever através dos arquivos de configuração apropriados por meio do componente *Configuration Loader*.

2.6. Configuration Loader

O *Configuration Loader* é carregado por um arquivo de configuração (no padrão XML), cuja estrutura já está previamente definida, e onde os requisitos de cada aplicação poderão ser expressos para definir quais eventos desejam receber (e.g. eventos relacionados a *switches*, *hosts*, *links* etc.).

3. Avaliação da proposta NOA

No intuito de validar os conceitos da proposta NOA, 4 (quatro) conjuntos de avaliação (sendo cada um deles em topologias de tamanhos arbitrários) foram conduzidos. Para cada um dos cenários foram considerados a execução de 3 (três) procedimentos básicos de rede: (i) inicialização (*bootstrap*); (ii) descoberta de rotas (realizada por meio do envio de subsequentes mensagens ICMP de *hosts* diversos presentes na topologia); e (iii) desligamento da topologia (*tear down*).

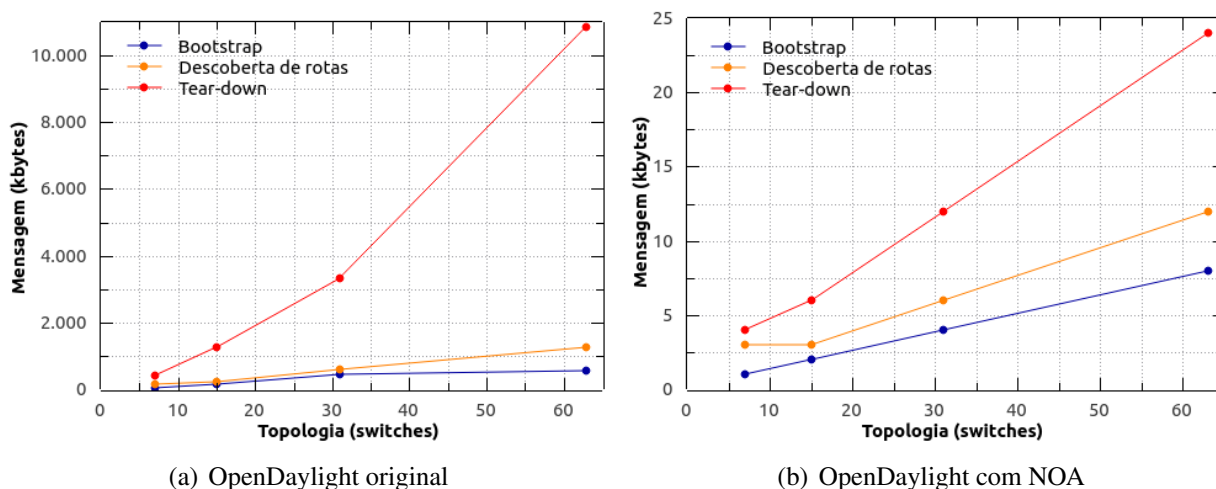


Figura 3. Resultados das avaliações

Toda a avaliação foi realizada por meio da virtualização da infraestrutura através da ferramenta Mininet⁵, da seguinte maneira: (i) **t1**, composta por 7 *switches*, 8 *hosts* e

⁵<http://mininet.org/>

14 links; (ii) **t2**, composta por 15 switches, 16 hosts e 30 links; (iii) **t3**, composta por 31 switches, 32 hosts e 62 links; e (iv) **t4**, composta por 63 switches, 64 hosts e 126 links. O objetivo da avaliação é validar a eficiência da proposta NOA em comparação as funcionalidades básicas providas pelo controlador OpenDaylight sem as extensões propostas.

Uma sumarização dos resultados é apresentado na Figura 3. Como exemplo, a Figura 3(a) demonstra que, para o processo de *bootstrap*, o OpenDaylight original enviou 53kb de dados para a topologia composta por 7 switches e 440kb para a topologia composta por 63 switches. Em contrapartida, conforme mostrado pela Figura 3(b), o OpenDaylight com o NOA, realizou o envio de apenas 1kb de mensagens para o *bootstrap* da topologia com 7 switches e 8kb para a topologia de 63 switches.

4. Conclusão

Neste trabalho propomos o *Network Orchestration Agent* (NOA), um *middleware* para maximização das capacidades ubíquas de orquestração do controlador OpenDaylight. NOA provê um conjunto de funcionalidades capazes de otimizar os mecanismos de orquestração desse controlador, viabilizando novos recursos que fornecem maior nível de abstração e ubiquidade para o desenvolvimento de novos serviços e aplicações para o gerenciamento das infraestruturas.

Até onde sabemos (e confirmado por estudos em trabalhos relacionados), não há evidências de propostas anteriores que forneçam um meio de notificação, em tempo de execução, sobre os eventos da topologia de forma granular. A próxima etapa deste trabalho consistirá em avaliar a eficácia do NOA em larga escala, por meio de *testbeds* experimentais (e.g. FIBRE).

Referências

- Boucadair, M. and Jacquenet, C. (2014). Software-Defined Networking: A Perspective from within a Service Provider Environment. Internet Engineering Task Force Request for Comments (RFC 7149).
- Enns, R., Bjorklund, M., Schoenwaelder, J., and Bierman, A. (2011). Network Configuration Protocol (NETCONF). RFC 6241, RFC Editor. <http://www.rfc-editor.org/rfc/rfc6241.txt>.
- Fielding, R. T. and Taylor, R. N. (2000). *Architectural styles and the design of network-based software architectures*, volume 7. University of California, Irvine Doctoral dissertation.
- Gomes, C. S., Silva, F. S. D., Neto, E. P., Costa, K. B., and da Silva, J. B. (2016). Towards a modular interactive management approach for sdn infrastructure orchestration. In *2016 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, pages 1–6.
- Medved, J., Varga, R., Tkacik, A., and Gray, K. (2014). Opendaylight: Towards a model-driven sdn controller architecture. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2014 IEEE 15th International Symposium on a*, pages 1–6. IEEE.
- ONF, O. N. F. (2014). SDN Migration Considerations and Use Cases. Technical report, Technical Report ONF TR-506, Open Network Foundation.