Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN

Felipe A. Lopes
Instituto Federal de Alagoas (IFAL)
Autonomic Networking Management


LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN

NMRG @ IETF 114
Challenges

Opportunities

How to enable autonomic networking in SDN?


LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN

NMRG @ IETF 114
Autonomic Networking Management in SDN

Challenges → Opportunities
How to enable autonomic networking in SDN?

Business Goals / Policies
Knowledge Generation / Analysis
Policy Processing
Configuration
Information Processing
Context

Control Plane

Research Challenges in Coupling Artificial Intelligence and Network Management
Jérôme François, Alexander Clemm, Dimitri Papadimitriou, Stenio Fernandes, Stefan Schneider


LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN
RFC 7575

IETF’s Network Management Group (NMRG)

Guidelines and Reference Model for implementing autonomic networks

Objective: Achieve Self-Management

Scope:

Self-* properties
- Self-configuration
- Self-healing
- Self-protection
- Self-optimizing

Design goals
- Coexistence with traditional management
- Decentralization
- Distribution
- Simplification of NBI
- Abstraction
- Autonomic monitoring
- ...
- 5 more

LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN
How to enable autonomic networking in SDN?

- Autonomic Network Management is not a new area
- Control Plane is software
- First papers from 2000’s
- IETF’s Community made contributions
- Automated Software Engineering Community made contributions
- Models at Runtime
- RFC 7575

LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN
The use of high-level models and formalisms in other computing areas

• **Software Engineering**: class diagram, UML, sequence diagram, components diagram...

• **Databases**: UML, entity relationship diagram, SQL generators...

• **Machine Learning**: components models and dataflows, neural networks models, black-box/trained models...

• **Networks**: topologies, NETCONF/YANG (is it easy to visualize? how could we define intents? how to grant correctness and the separation of responsibilities?), how to enable zero/one-touch from these models?
The word “Models” of MART comes from the Model-Driven Engineering (MDE) discipline.

Models at Runtime (MART)

The word “Models” of MART comes from the Model-Driven Engineering (MDE) discipline.

- Objective/Adaptation Models
- Learning/Monitoring
- Reasoning
- Analyzer
- Executor
- Managed System

Proposed MART Architecture

**RFC 7575 Design Goals**

- Self-configuration
- Self-healing
- Self-optimizing
- Abstraction
- Autonomic Reporting
- Independence of Function and Layer
- Full Life-Cycle Support

LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN
RFC 7575 and MART for Autonomic Networking in SDN (and related I-Ds)

LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN
RFC 7575 and MART for Autonomic Networking in SDN

Input: Retrieve data from the K-Base; if exists
Input: Initialize weights for action-value pairs, state $s_t$, objective $g$;
Data: reached = false;
Data: episode = 0;
begin
   while !reached do
      for step < learning_rate; step++ do
         Select action $a_t$ in $s_t$;
         Query network parameters in $s_t$;
         Select a parameter $param_t$;
         $e = e - (step/learning_rate) * e$;
         Execute action $a_t$ for $param_t$;
         Observe new state $s_{t+1}$;
         Calculate reward $r_t$ and store experience in K-Base;
         Update $param_t$ in the Planned Model;
         Collect samples of $n$ randomized transitions of K-Base;
         Calculate discount factor $d$ of the reward $r_t$;
         Update the transition $t_q$ with $d$;
         Train the Q-Network with the new $d$ values of transition $t_q$, $s_{t+1}$, and $a_t$;
         reached = compare($g$, better, worst)
      end
      episode++;
   end

Development and Management Models as inputs

Running code / Network rules as outputs
The conceptual differences between Intent, Policy, and Service models are clear at the draft:

- “Intent-Based Networking - Concepts and Definitions” (Alexander Clemm, Laurent Ciavaglia, Lisandro Zambenedetti Granville, Jeff Tantsura)
- Is the proposed MART approach a feasible way to implement these abstractions?

Could the Autonomic Resource Control Architecture (ARCA) (from the draft entitled “Artificial Intelligence Framework for Network Management”) benefit from a new abstraction layer?

- How intents will be “translated” into network rules? And how they will be integrated with the monitoring and adaption actions?
  - A MART-based definition may help
Conclusions

Proposal for enabling autonomic network management in SDN

Knowledge combination

High-level modeling for implementing autonomic behavior

Some design goals could not be covered

Discussion

Proper functioning and suitable performance
Acknowledgement
Thank you! Questions?
<felipe.alencar@ifal.edu.br>
Backup slides
Evaluation

Which RFC 7575’s design goals the proposed MART-based solution could not achieve? Why?

How suitable is the proposed MART-based solution from a performance perspective?
Evaluation

Which RFC 7575’s design goals the proposed MART-based solution could not achieve? Why?

How suitable is the proposed MART-based solution from a performance perspective?
Evaluation

*Which RFC 7575’s design goals the proposed MART-based solution could not achieve? Why?*

<table>
<thead>
<tr>
<th>RFC 7575 Design Goals not achieved</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self-protection</td>
<td>Currently, metamodels do not have entities for representing such goal.</td>
</tr>
<tr>
<td>Coexistence with Traditional Management</td>
<td>The final ANM system supports only intents.</td>
</tr>
<tr>
<td>Secure by Default</td>
<td>The final ANM system cannot assert the membership of all components as required by the RFC.</td>
</tr>
<tr>
<td>Common Autonomic Networking Infrastructure</td>
<td>Achieving this goal using a unique MART solution would increase the complexity of metamodels.</td>
</tr>
</tbody>
</table>
Evaluation

How suitable is the proposed MART-based solution from a performance perspective?

Metamodels and algorithms implemented in Graphical Modeling Framework (GMF)

Use case

Modeling

Simulation

Enable the communication in a network topology consisting of three nodes connected by four switches, forming two different paths. All the links have 10Mb of bandwidth.

https://github.com/felipealencar/mdn

LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN
How suitable is the proposed MART-based solution from a performance perspective?

Use case

Modeling

Configuration Model

Objectives Model

LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN
How suitable is the proposed MART-based solution from a performance perspective?

Models as inputs of code templates

Modeling

Simulation

Code Generation

Monitoring

Learning

Mininet network simulation

LOPES, F. A., Using the RFC 7575 and Models at Runtime for Enabling Autonomic Networking in SDN
How suitable is the proposed MART-based solution from a performance perspective?

Select actions and network monitoring to check if h1 reaches h2.

(a) Learning rate for the use case.

Learning rate: 0.0001
Activation function: ReLu
How suitable is the proposed MART-based solution from a performance perspective?

Introduce congestion traffic to verify model transformations and reasoning in the action selection process.

(b) Delay rate.