Using Automated Planning in data centers fault tolerance systems

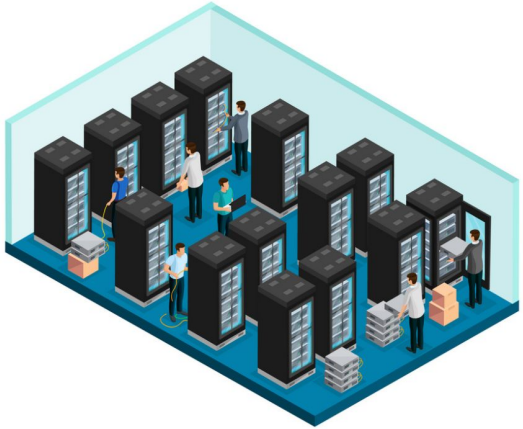
Douglas Trajano
Master's Degree in Computer Science
Pontifical Catholic University of Rio Grande do Sul - PUCRS

Problem Statement

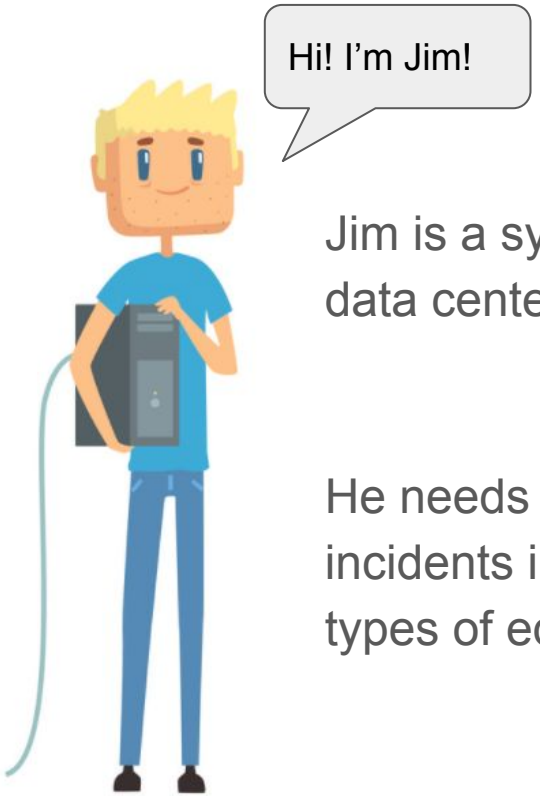
Data centers are complex systems, its components are fully integrated and an impact in one of these components can affect other components.

Engineers need to solve critical incidents in timely fashion.

Knowledge bases are built to help these engineers to follow a plan for fixing known issues.



Let’s know Jim!



Jim is a system administrator in a data center.

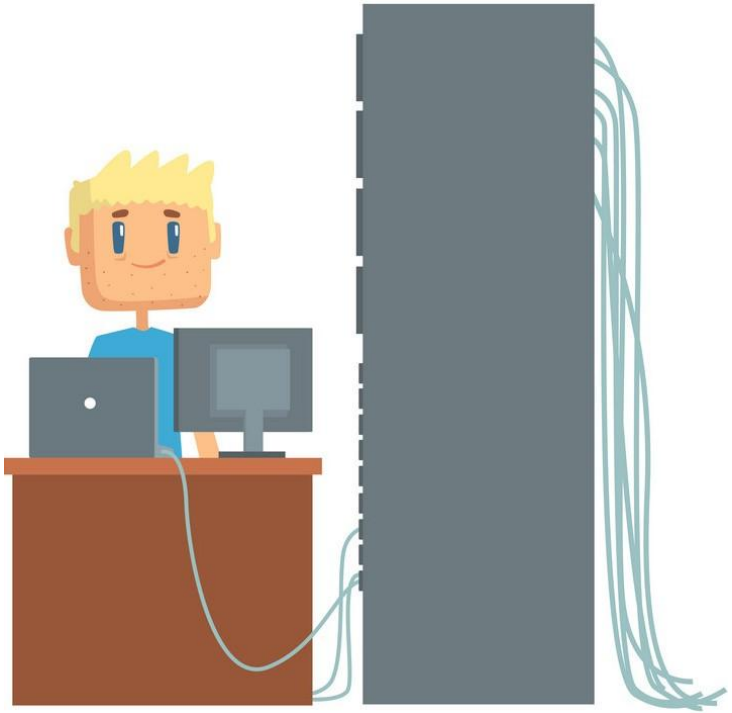
He needs our help to solve critical incidents in a variety of different types of equipment.

Proposed solution

With **PDDL**, we can formalize domains and problems simulating known issues in our infrastructure.

The **automated planner** provides a set of ordered actions that we need to do to solve the incidents or apply a workaround solutions.

It can act directly in the infrastructure or can be used as a guide to help Jim.

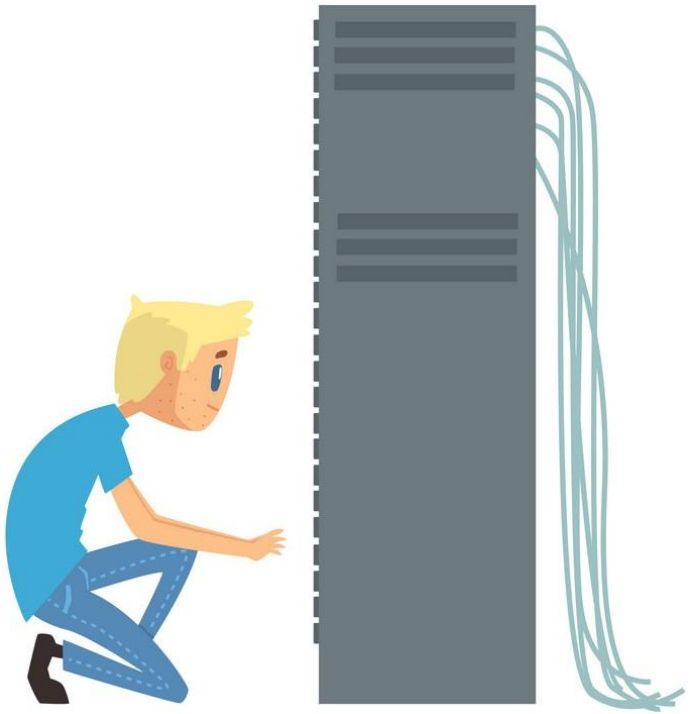


Experiments

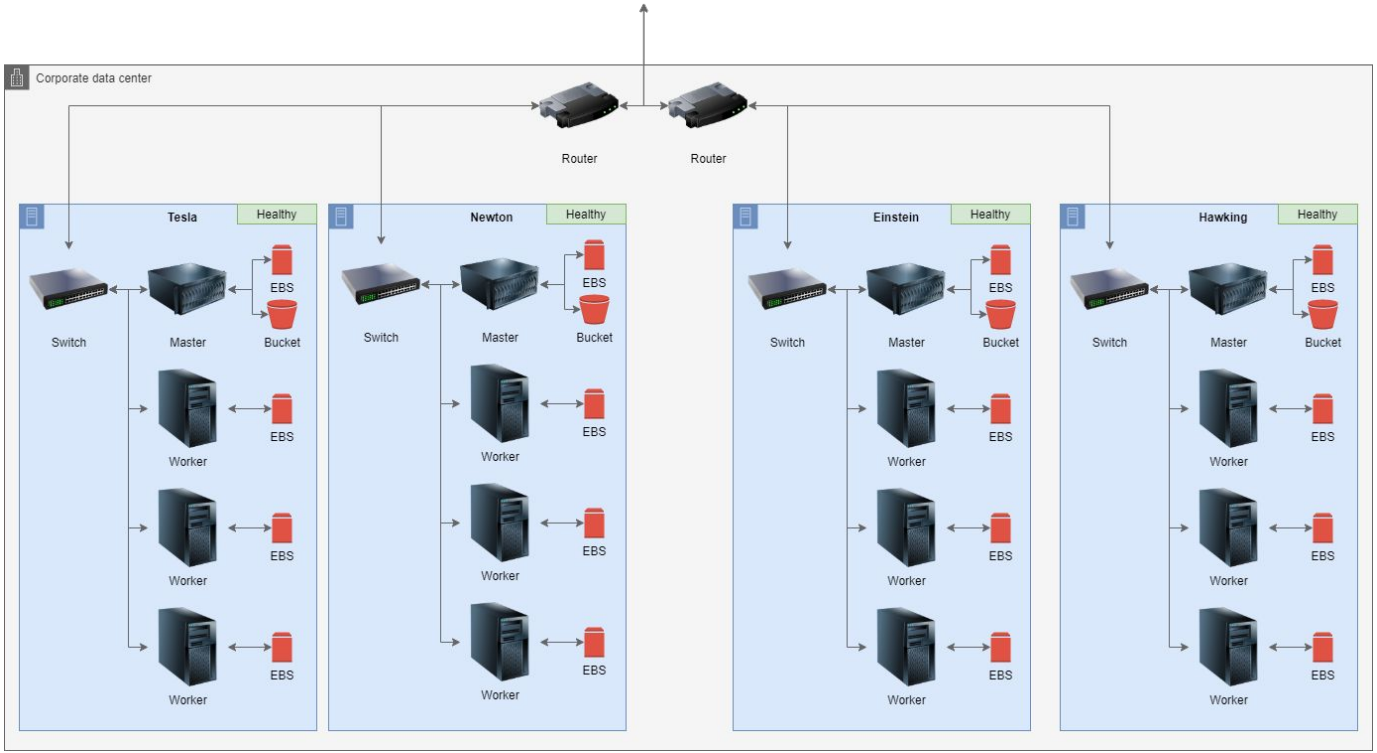
1 domain and 3 problems (scenarios) are defined in this work.

The **initial states** represent a known issue in our infrastructure.

The **goal** for all the problems is the same. All clusters and components with a healthy state.



Experiments - Domain



Router

Establish a connection between clusters (switches) and the external world.

Predicates

- router-healthy
- router-pair

Actions

- router-turn-on
- router-turn-off
- router-mesh-on
- router-mesh-off

Switch

Connect all components in a cluster.

Predicates

- switch-healthy
- switch-attach-router
- switch-attach-master
- switch-attach-worker

Actions

- switch-turn-on
- switch-turn-off

Master computer

A controller inside the cluster, it provides server tools to worker computers.

Predicates

- master-healthy

Actions

- master-turn-on
- master-turn-off

Worker computer

Provide computational power, applications are deployed in worker computers.

Predicates

- worker-healthy
- worker-high-mem
- worker-high-cpu
- worker-high-network

Actions

- worker-turn-on
- worker-turn-off

EBS

It works as a HDD (hard disk drive) for computers.

Predicates

- ebs-healthy
- ebs-locked
- ebs-attached

Actions

- ebs-turn-on
- ebs-turn-off

Bucket

Object storage. It is used to provide a shareable storage between applications.

Predicates

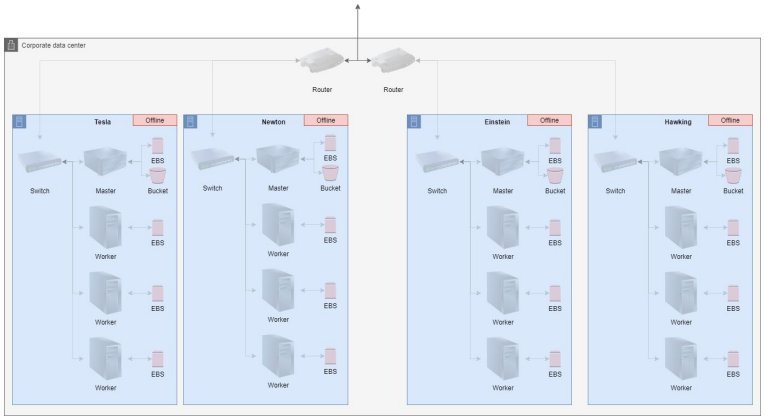
- bucket-healthy
- bucket-locked
- bucket-attached

Actions

- bucket-turn-on
- bucket-turn-off

Experiments - Problem 1

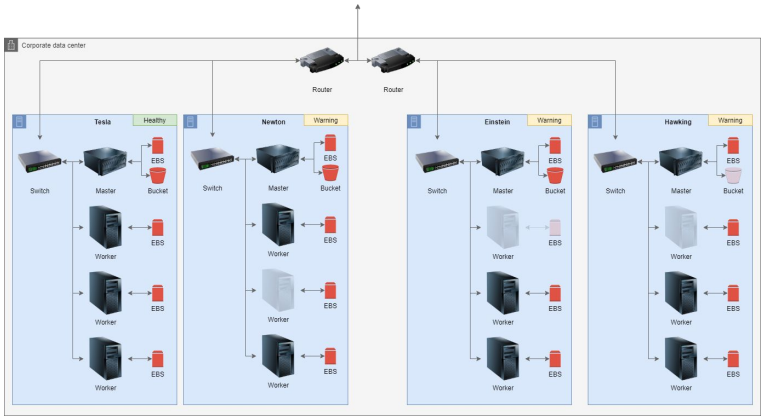
Problem 1 explores the initial state of our data center where all components are offline.

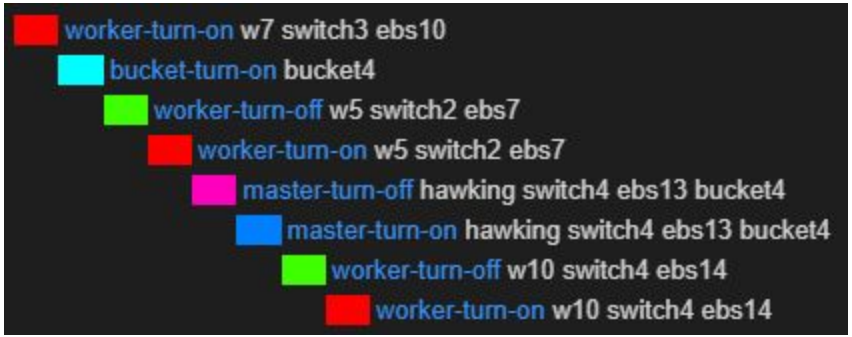



```
router-turn-on router1
switch-turn-on switch1 router1
switch-turn-on switch2 router1
switch-turn-on switch3 router1
switch-turn-on switch4 router1
router-turn-on router2
router-mesh-on router1 router2
ebs-turn-on ebs1
worker-turn-on w12 switch4 ebs1
ebs-turn-on ebs2
worker-turn-on w11 switch4 ebs2
ebs-turn-on ebs3
worker-turn-on w10 switch4 ebs3
ebs-turn-on ebs4
worker-turn-on w9 switch3 ebs4
ebs-turn-on ebs5
worker-turn-on w8 switch3 ebs5
ebs-turn-on ebs6
worker-turn-on w7 switch3 ebs6
ebs-turn-on ebs7
worker-turn-on w6 switch2 ebs7
ebs-turn-on ebs8
worker-turn-on w5 switch2 ebs8
ebs-turn-on ebs9
worker-turn-on w4 switch2 ebs9
ebs-turn-on ebs10
worker-turn-on w3 switch1 ebs10
ebs-turn-on ebs11
worker-turn-on w2 switch1 ebs11
ebs-turn-on ebs12
worker-turn-on w1 switch1 ebs12
ebs-turn-on ebs13
ebs-turn-on ebs14
ebs-turn-on ebs15
ebs-turn-on ebs16
bucket-turn-on bucket1
master-turn-on hawking switch4 ebs13 bucket1
bucket-turn-on bucket2
master-turn-on einstein switch3 ebs14 bucket2
bucket-turn-on bucket3
master-turn-on newton switch2 ebs15 bucket3
bucket-turn-on bucket4
master-turn-on tesla switch1 ebs16 bucket4
```

Experiments - Problem 2

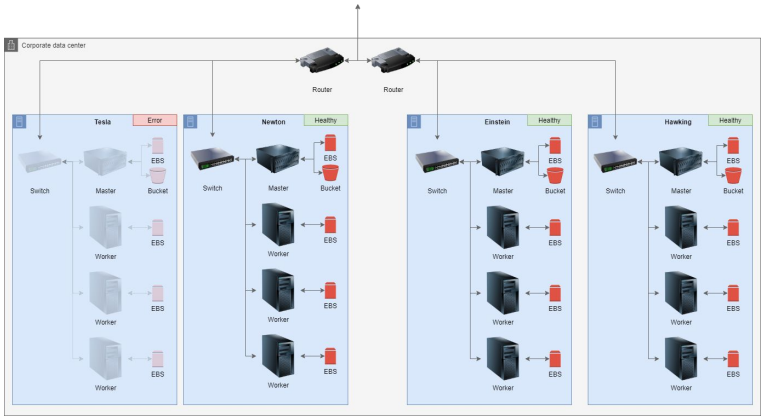
Here we have some components with errors in three different clusters.





Experiments - Problem 3

An entire cluster here is offline, we want a plan with actions to turn on all components in Tesla cluster.



switch-turn-on switch1 router1
ebs-turn-on ebs1
worker-turn-on w3 switch1 ebs1
ebs-turn-on ebs2
worker-turn-on w2 switch1 ebs2
ebs-turn-on ebs3
worker-turn-on w1 switch1 ebs3
ebs-turn-on ebs4
bucket-turn-on bucket1
master-turn-on tesla switch1 ebs4 bucket1

Conclusions

The idea of using PDDL to formalize domains of data centers is very interesting, but there are still many questions to be answered.

- Is it reasonable to require PDDL knowledge for engineers? Can engineers develop and maintain the domain and problem definitions?
- How the information represented by predicates will be collected for the planner?
- Can we allow the planner to act directly on the infrastructure? Should some actions be restricted to engineers?

Future work

As the future work, is interesting to explore more complex architectures, with more components, involving applications and more possible situations (problems).

Define more appropriate actions for the components will provide better plans avoiding a sequence of the turn off and turn on that ca be stressful for the system.