A Software Tool for Dependability and Cost Evaluation of Private Cloud Infrastructures

Edson Samuel Jr
esgsj@cin.ufpe.br
Context

- Cloud Computing Infrastructure Planning
- How do we get better solutions amongst a large combination of cloud infrastructure components?
- Combined values: component types and redundancy
- Decision factors: dependability and cost
Objective

Development of a software tool with the purpose to conceive and evaluate private cloud computing environments considering cost, and dependability aspects.
Specific Objectives

● Suggest solutions for cloud infrastructure arrangement considering dependability and cost constraints
● Present an efficient methodology for modeling of cloud components representation
Context

Component Types (Eucalyptus-based IaaS)

- Cloud Controller (CLC)
- Cluster Controller (CC)
- Node Controller (NC)
- Virtual Machine (VM)
- Network Switch (SW)
- Router (RTR)
Context

Redundancy Types

- **Active - Passive**
  - Hot-Standby
  - Cold-Standby
  - Warm-Standby

- **Active - Active**
  - N + 1

- **No Redundancy**
Restrictions

User-based definitions of constraints for classification of solutions

- Cost value (US$)
- Availability rate (%)
Model Generation

- Greedy Randomized Adaptive Search Procedure
  (Resende & Ribeiro, 2002)

- Comprises
  - Construction phase
  - Local search phase
Model Generation

- Our algorithm is an adaptation for the original GRASP metaheuristic definition

```plaintext
procedure GRASP(Max_Iterations,Seed)
1    Read_Input();
2    for k = 1,...,Max_Iterations do
3        Solution ← Greedy_Randomized_Construction(Seed);
4        Solution ← Local_Search(Solution);
5        Update_Solution(Solution,Best_Solution);
6    end;
7    return Best_Solution;
end GRASP.
```
A pivot solution is created and used as a reference model (seed) for the construction of the solution set.
Model Generation

Local Search Phase

Picks the best solution on a defined criteria and sets it up as a new seed.
Algorithm

- In the end of a local search, the algorithm provides a optimized solution set
- Restricted Candidate List
- Performing this procedure for a maximum number of iterations will yield a “elite” solution set
- That’s what we want
DCM4PCIP’s Case Study

- Dependability and Cost Modeling for Private Cloud Infrastructure Planning
- Developed using Java Platform
- Uses Mercury Tool
  - Model Generation API
  - Stationary simulation for availability metric computation
Communication Workflow

ModelGenerator

Generate

RemoteEvaluatorClient

Send

Receive

MercuryInstance

Evaluate

Input

Output

opt

Return

[ (cost < x) && (availability > y) ]

[ else ]

Regenerate

Send

Receive

Evaluate