SDN for Inter Cloud Networking

Marouen Mechtri, Ines Houidi, Wajdi Louati, **Djamal Zeghlache**
Institut Mines-Télécom, Télécom SudParis, Evry, France

Objectives

- **Cloud networking**
  - Provide connectivity in a non-intrusive way of distributed resources (preserves the network configuration of cloud providers) .
  - Establish dynamically and control end to end connectivity of distributed resources and services
  - Cloud and network providers involved

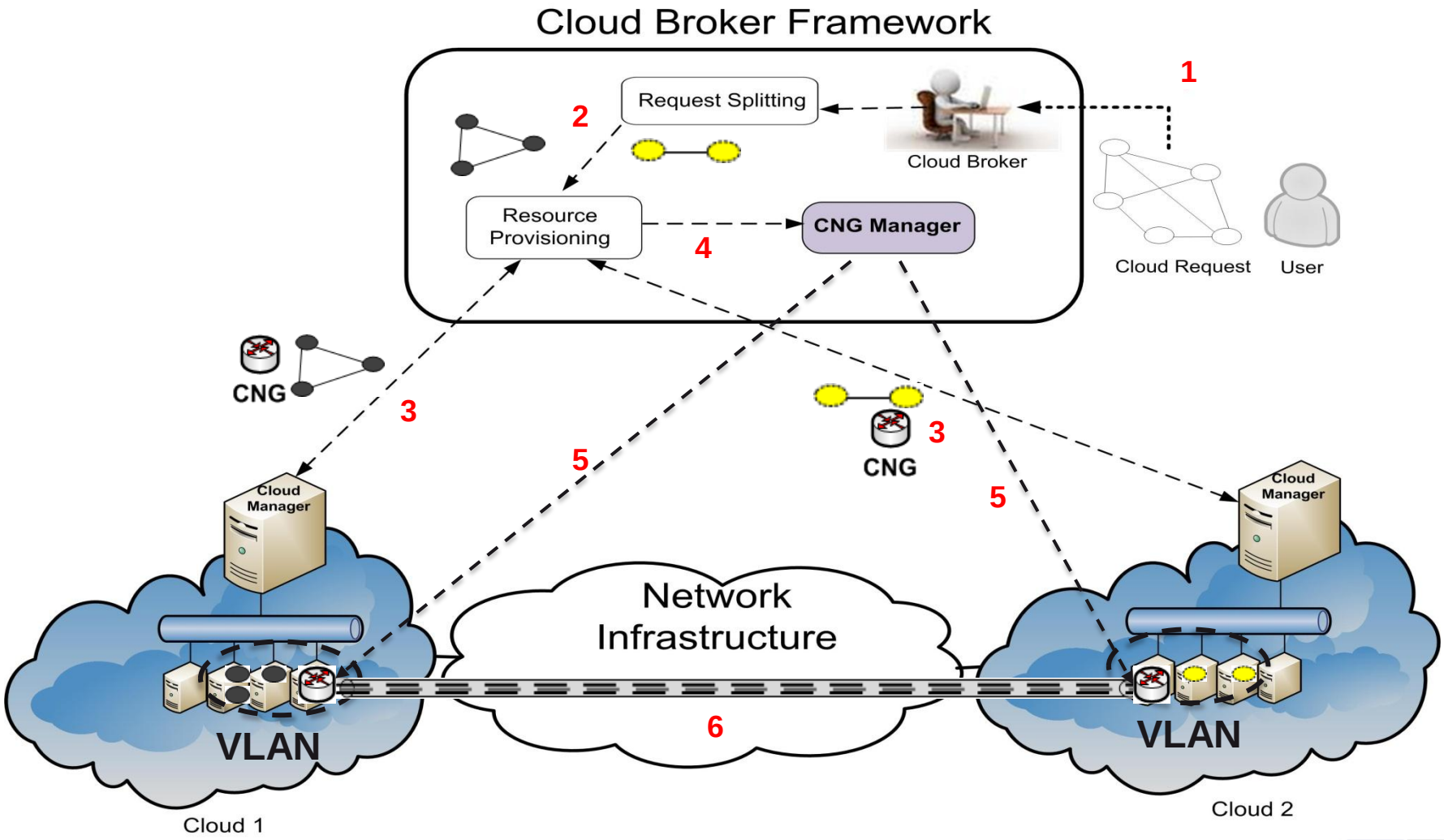
Summary

- Cloud Broker Framework
- Cloud Request Splitting
- Cloud Networking Gateway Manager description
- Performance Results

Contents

- **Cloud Broker Framework**
- Cloud Request Splitting
- Cloud Networking Gateway Manager description
- Performance Results

Cloud Broker Framework

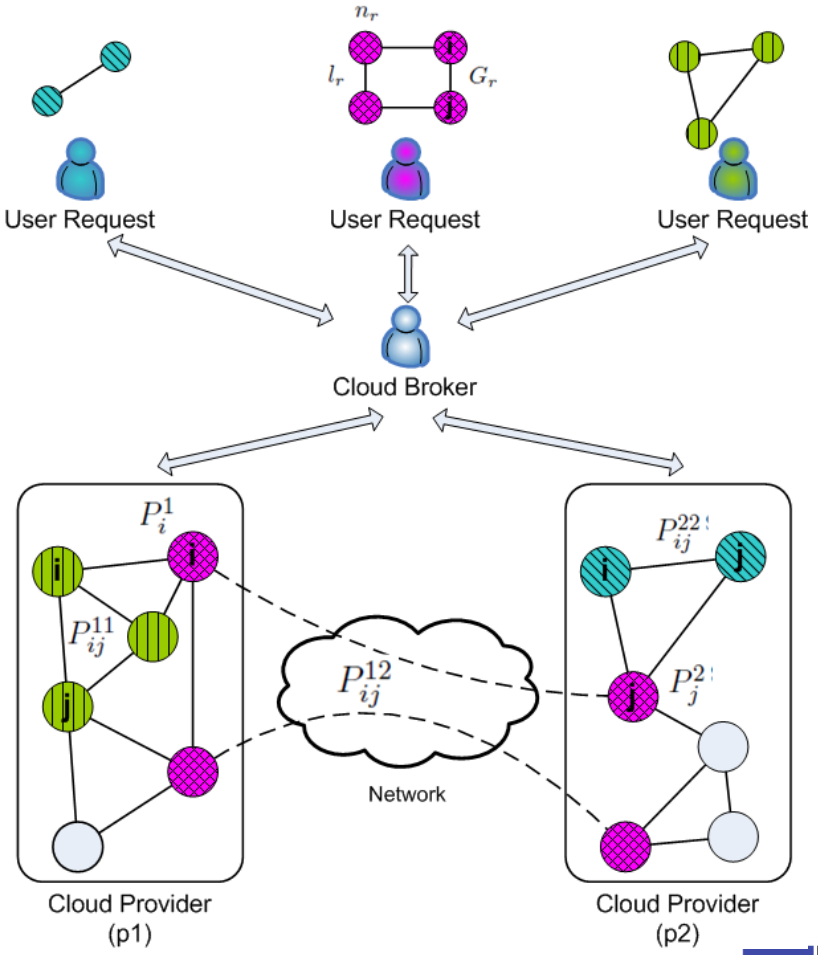


Contents

- Cloud Broker Framework
- **Cloud Request Splitting**
- Cloud Networking Gateway Manager
- Performance Results

Cloud Request Splitting

- **Objective:**
  - Find optimal (minimum connectivity cost) request splitting across multiple cloud providers
  - Splitting algorithm determines the sub-graphs to providers mapping



Cloud Request Splitting

- x_i^j : set to 1 if resource i is associated with cloud provider j .
- $y_{ii'}^{jj'}$: set to 1 if a requested link between nodes $i, i' \in N_r$ involves cloud providers j and j'.
- P_i^j : price (or cost) of requested node i if provided by cloud provider cloud j.
- $P_{ii'}^{jj'}$: price (or cost) of link between, nodes $i, i' \in N_r$ when involving cloud providers j and j'

- **Quadratic program formulation:**

$$(A) \begin{cases} min(\sum\limits_{i \in N_r} \sum\limits_{j \in Pr} P_i^j x_i^j + \sum\limits_{\substack{(i,i') \in N_r \\ i<i'}} \sum\limits_{(j,j') \in P_r} P_{ii'}^{jj'} x_i^j x_{i'}^{j'}) \\ \sum\limits_{j \in P_r} x_i^j = 1; \quad \forall i \in N_r; \quad x_i^j \in \{0,1\} \end{cases}$$

- **Linear integer program formulation:**

$$(A) \Leftrightarrow \begin{cases} min(\sum\limits_{i \in N_r} \sum\limits_{j \in Pr} P_i^j x_i^j + \sum\limits_{\substack{(i,i') \in N_r \\ i<i'}} \sum\limits_{(j,j') \in P_r} P_{ii'}^{jj'} y_{ii'}^{jj'}) \\ \sum\limits_{j' \in P_r} y_{ii'}^{jj'} = x_i^j; \quad \forall i, i' \in N_r; \ \forall j \in Pr \\ x_i^j + x_{i'}^{j'} - y_{ii'}^{jj'} \leq 1 \\ \sum\limits_{j \in P_r} x_i^j = 1; \quad \forall i \in N_r; \quad x_i^j \in \{0,1\}; \quad y_{ii'}^{jj'} \in \{0,1\} \end{cases}$$

Contents

- Cloud Broker Framework
- Cloud Request Splitting
- **Cloud Networking Gateway Manager description**
- Performance Results

Cloud Networking Gateway Manager

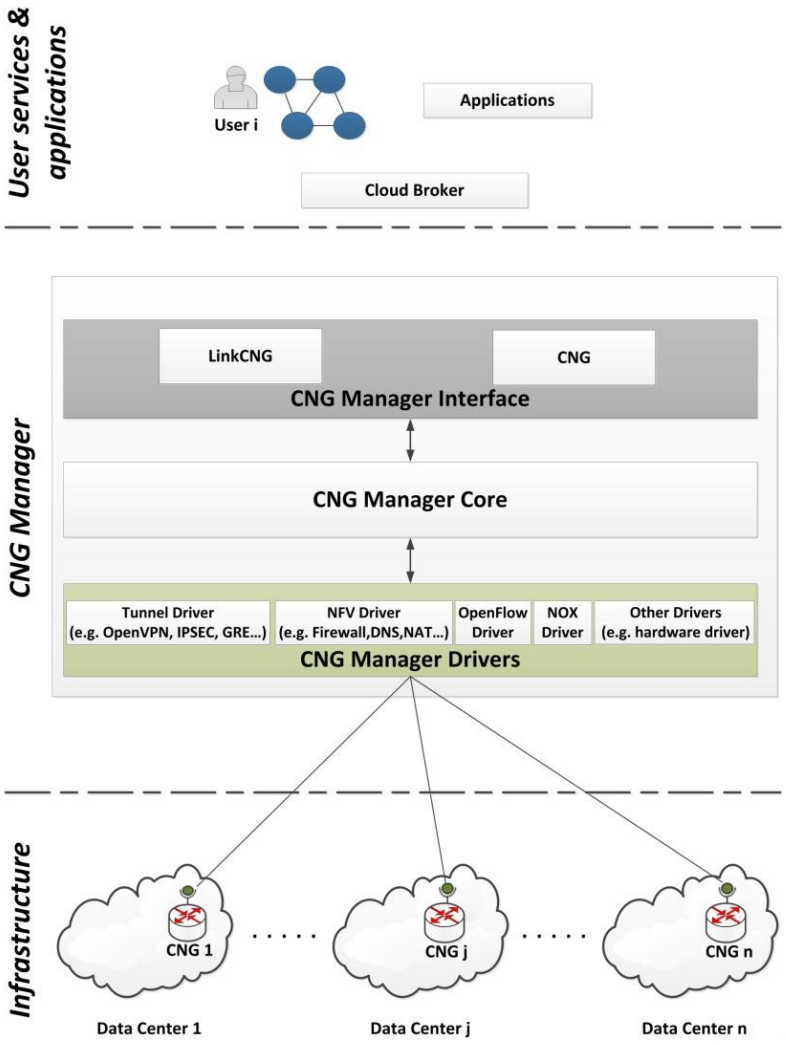
- **The CNG-M in the proposed networking architecture can be seen as an SDN controller**
- **The networking architecture relies on two main components:**
  - SDN controller called Cloud Networking Gateway Manager (CNG Manager)
  - A virtual and generic appliance (CNG)
    - acting as a gateway between user resources (named Cloud Networking Gateway, CNG).
    - the CNG provides a RESTful interface to enable the configuration and the programmability of its features by the CNG Manager.

Cloud Networking Gateway Manager

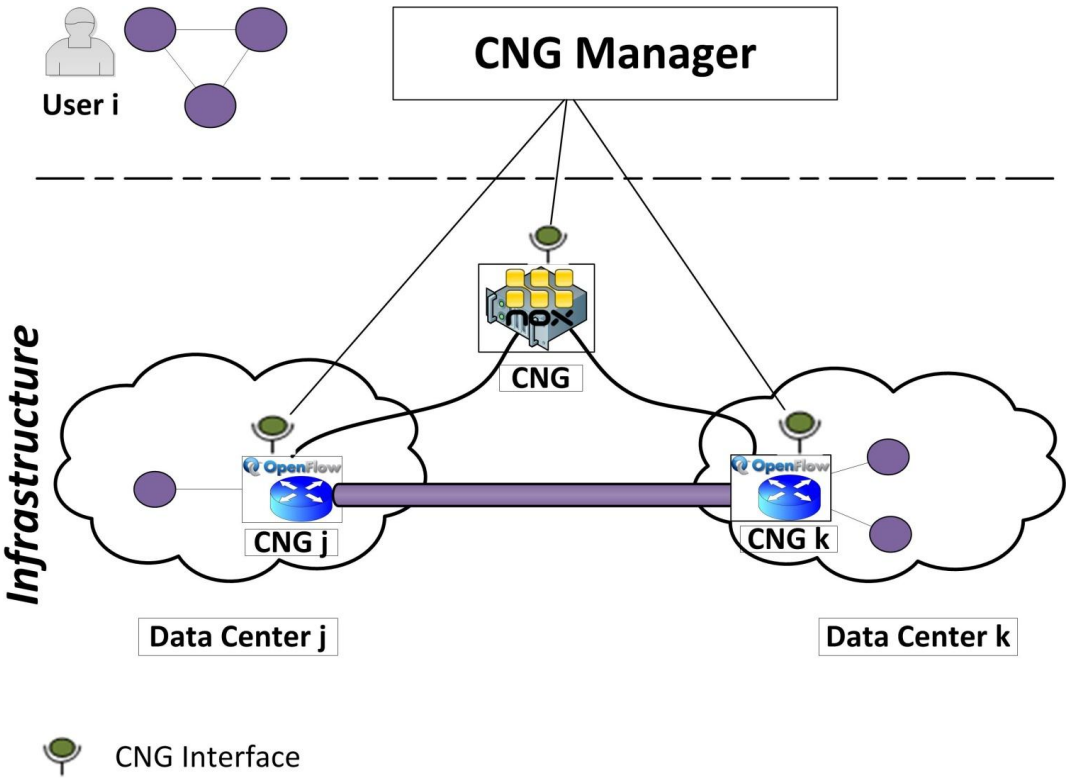
- **Ensures connectivity between resources acquired from distributed and independent cloud providers.**
- **Gives partial or complete control of connectivity to the users so they network their applications.**

Cloud Networking Gateway Manager

- **The CNG Manager has:**
  - Northbound interface towards client requesting connectivity based on the OCCI specification and service model.
  - The CNG Manager Core selects the appropriate drivers, in line with user expressed networking requirements.
  - Southbound interface interacting with transport technologies through specific drivers.



Example with OpenFlow network



- CNG Manager relies on an OpenFlow driver to configure 2 CNGs acting as OpenFlow switches connected to a NOX controller

Contents

- Cloud Broker Framework
- Cloud Request Splitting
- Cloud Networking Gateway Manager
- **Performance Results**

Performance Results (CNG Manager)

Cloud Providers	2	4	6	8	10
Delay (s)	6.77	18.86	27.29	44.77	59.135

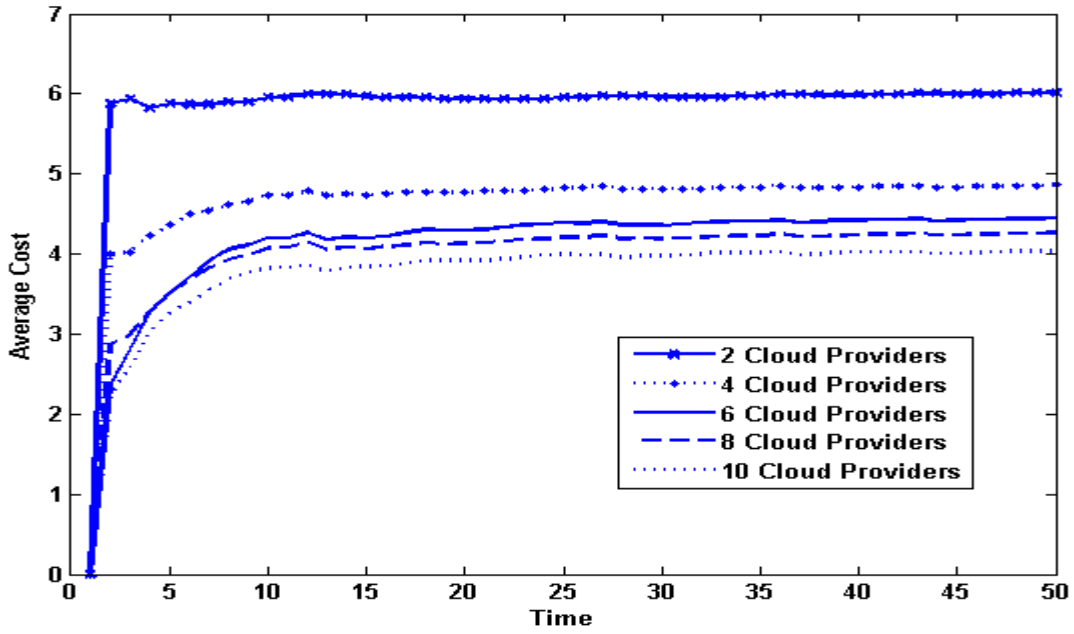
- Configuration delays of the network graphs composed through CNG nodes.
- Each node of the network graph represents a cloud provider.

Performance Results (Splitting algorithm)

Cloud Providers \ Request Size	5	10	15	17	20
2	0	0.01	0.02	0.02	0.09
3	0	0.03	0.08	0.05	0.36
4	0	0.05	0.33	0.47	0.67
5	0.01	0,08	0.55	0.81	1.61
6	0.01	0.23	0.94	2.38	4.05

- Delay required to split a Cloud request between multiple Cloud Providers: Exact method.

Performance Results (Splitting algorithm)



- **Average cost generated by the graph splitting algorithm when multiple Cloud Providers are involved.**

Conclusion

- Proposed an SDN controller (the CNG Manager) to achieve dynamic on demand inter cloud networking.
- Control of connectivity between distributed resources acquired from multiple cloud providers.
- The CNG Manager is available at:
  - *https://github.com/MarouenMechtri/CNG-Manager*
- Future work :
  - multi-objectives optimization as needed in the cloud context.
  - generalizing SDN principles to support distributed & connected clouds services.

INSTITUT Mines-Télécom

Thank you for your attention

Questions?

deux écoles filiales et deux partenaires stratégiques