Use cases for operating networks in the overlay model context

CCAMP WG, IETF88, Vancouver

draft-ceccadedios-ccamp-overlay-use-cases-04

Daniele Ceccarelli <daniele.ceccarelli@ericsson.com>

Oscar Gonzalez de Dios <ogondio@tid.es>

Fatai Zhang <zhangfatai@huawei.com>

Xian Zhang <zhang.xian@huawei.com>

J.Tantsura, D.Caviglia, K.Pithewan, C.Margaria,

J.Drake, S.Belotti, V.Lopez V.P.Beheram

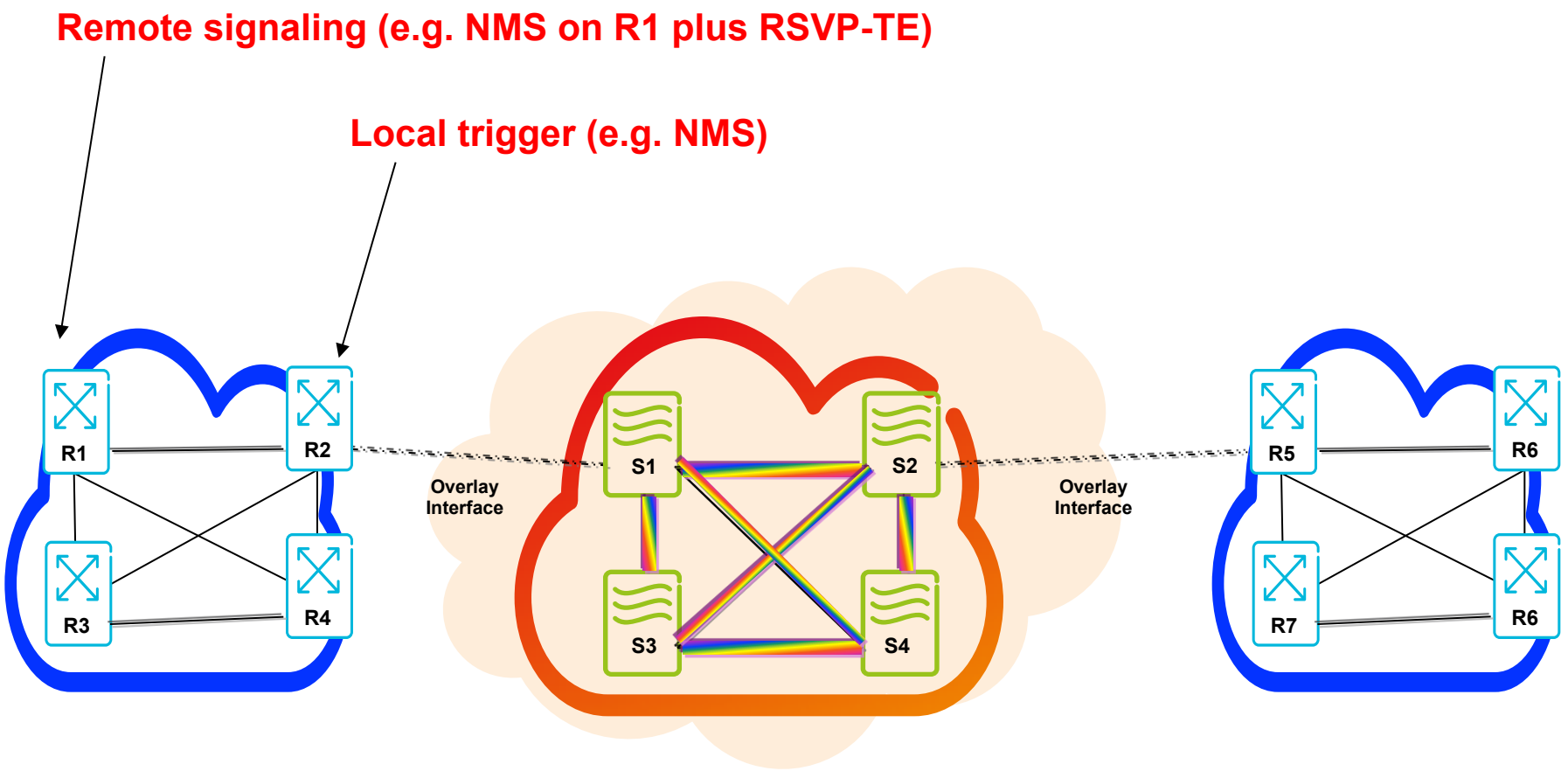
J.Meuric I.Bryskin G.Grammel

The draft would like to…

- …define use cases for operating overlay networks
- …define a set of assumptions to be used as the basis for the design of use cases
- …trigger discussion on which ones are needed and which ones are not

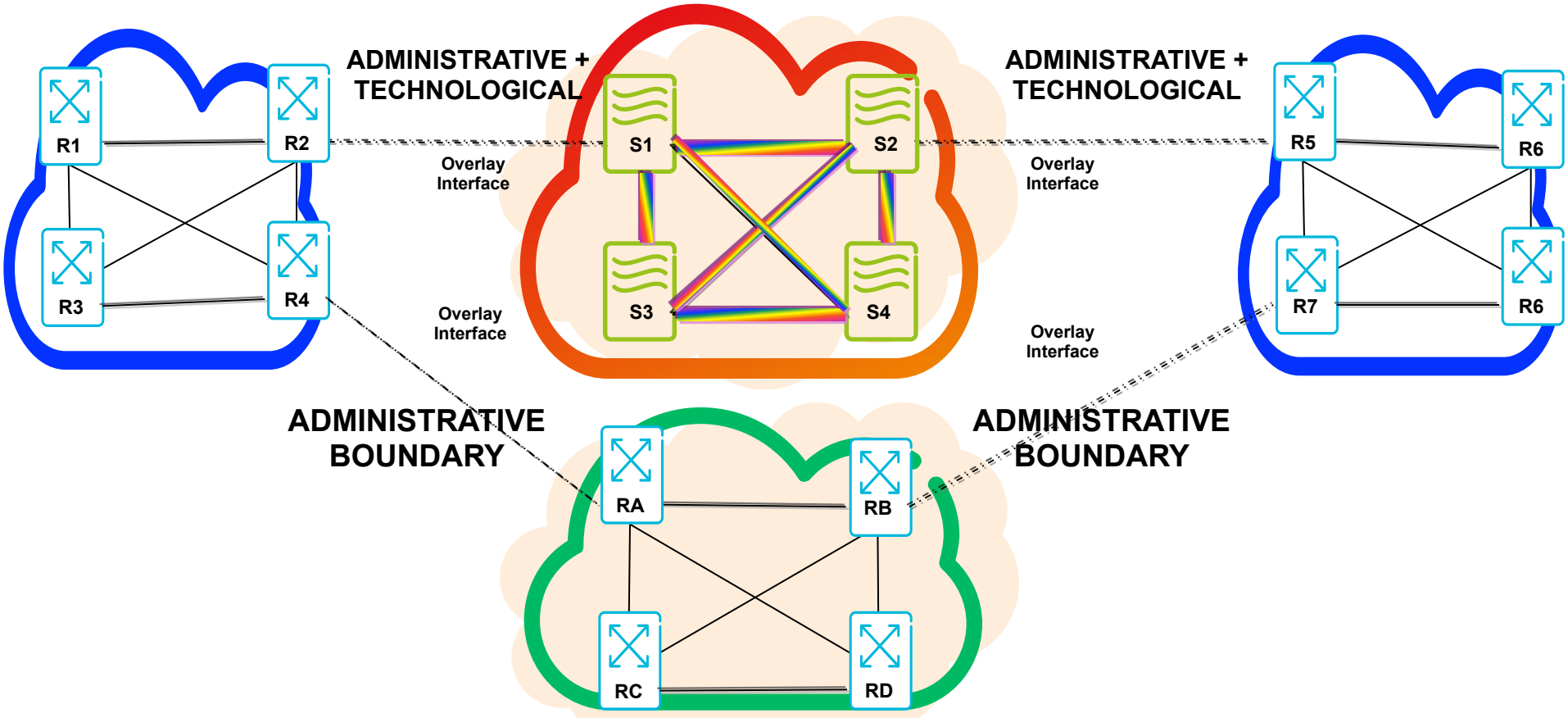
Terminology – Applies to all UCs

- 1. Local trigger vs remote signaling



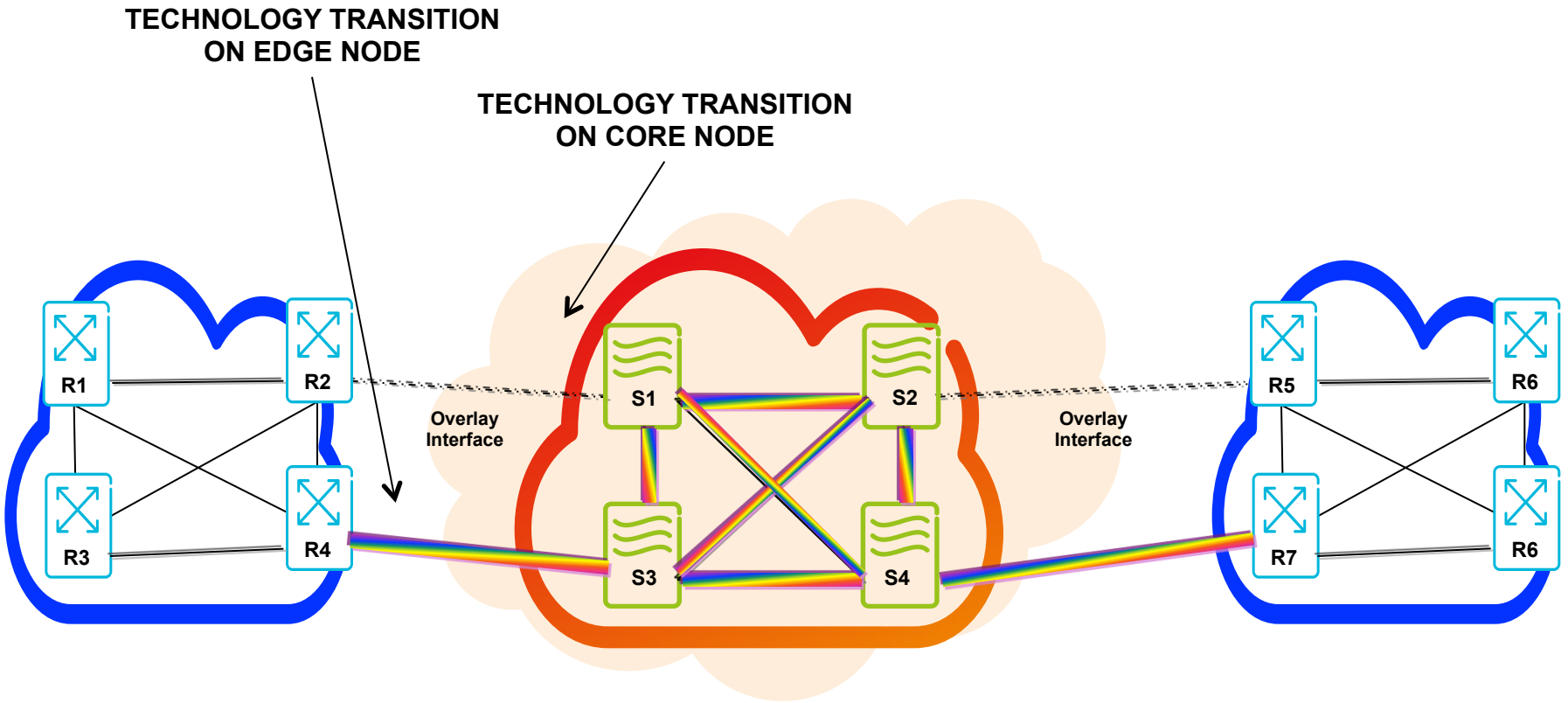
Terminology – Applies to all UCs

- 2. Administrative boundary vs Administrative and technological boundary



Terminology – Applies to all UCs

- 3. Technology transition on edge node vs Technology transition on core node



Use cases

- UC 1 – Provisioning

  Requirement: The network operator must be able to setup an unprotected end to end service between two client layer nodes.

- UC 2 - Provisioning with optimization

  Requirement: The network operator must be able to setup a service expressing which parameter must be optimized when computing the path. The server domain should tell the client domain what prevented a requested to be satisfied. Subsequent actions (e.g., use a different interface, relax constraints, send an alarm to the client domain's CEO...) are up to the client domain.

- UC 3 - Provisioning with constraints

  Requirement: The network operator must be able to setup a service imposing upper/lower bounds for a set of parameters during the path computation.

Use cases

- UC 4 – Provisioning with diversity

  Requirement: The N.O. must be able to setup a service in the server layer in diversity with respect to server layer resources or not sharing the same fate with other server layer services.

- UC 5 – Remote dual homing

  Requirement: The N.O. must be able to setup a plurality of services not necessarily between the same pair of edge nodes.

- UC 6 – Re-optimization

  Requirement: The network operator must be able to setup a service so that the overall cost of the network is minimized and not the cost of a single service.

(*) – No text yet or questioning on usefulness

Use cases

- UC 7 – Query

  Requirement: The server network must be able to tell the network operator the actual parameters characterizing an existing service.

- UC 8 – Availability Check

  Requirement: The network operator must be able to check if in the server layer there are enough resources to setup a service with given parameters.

- UC 9 – P2MP services

  Requirement: If allowed by the technology, the network operator must be able to setup a P2MP service with given parameters.

- UC 10 – Privacy

  Requirement: The network operator must be able to provision different groups of users with independent addressing spaces.

(*) – No text yet or questioning on usefulness

Use cases

- UC 12 – Stacking of overlay interfaces

  Requirement: The network operator must be able manage a network with an arbitrarily high number of administrative boundaries (i.e.,>2).

- UC 13 – Resiliency parameters

  Requirement: The network operator must be able to request an LSP in the server layer with resilience parameters. E.g., 1+1 protection and restoration.

  Moreover, it must be possible for the operator to change the resilience level after the path is established in the network.

- UC 14 – Inquiry (to be added)

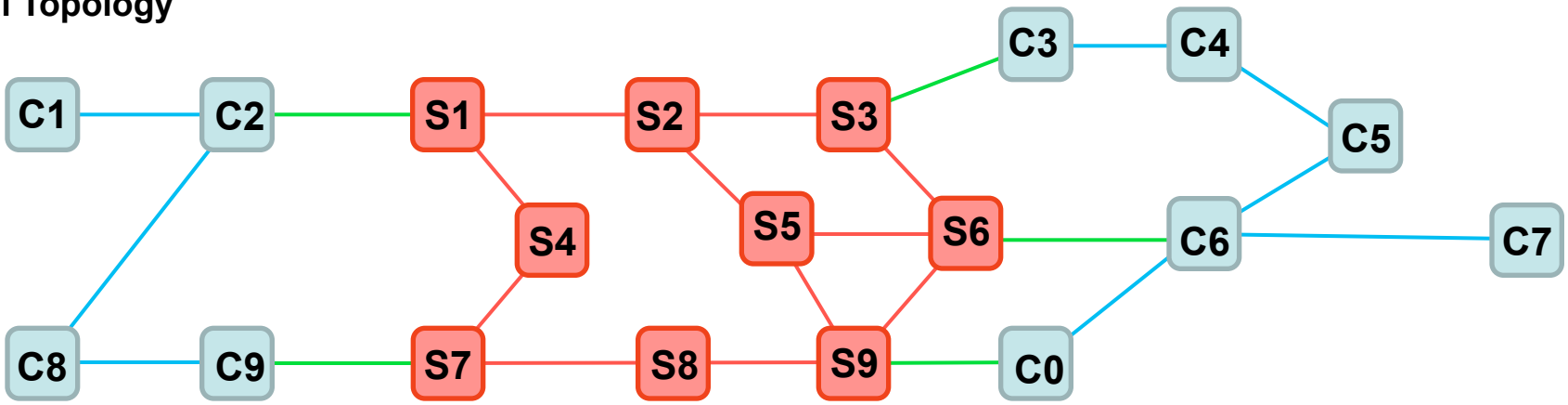
  Requirement: Client layer must be able to inquire server layer if a given service can be re-optimized.

Next Step

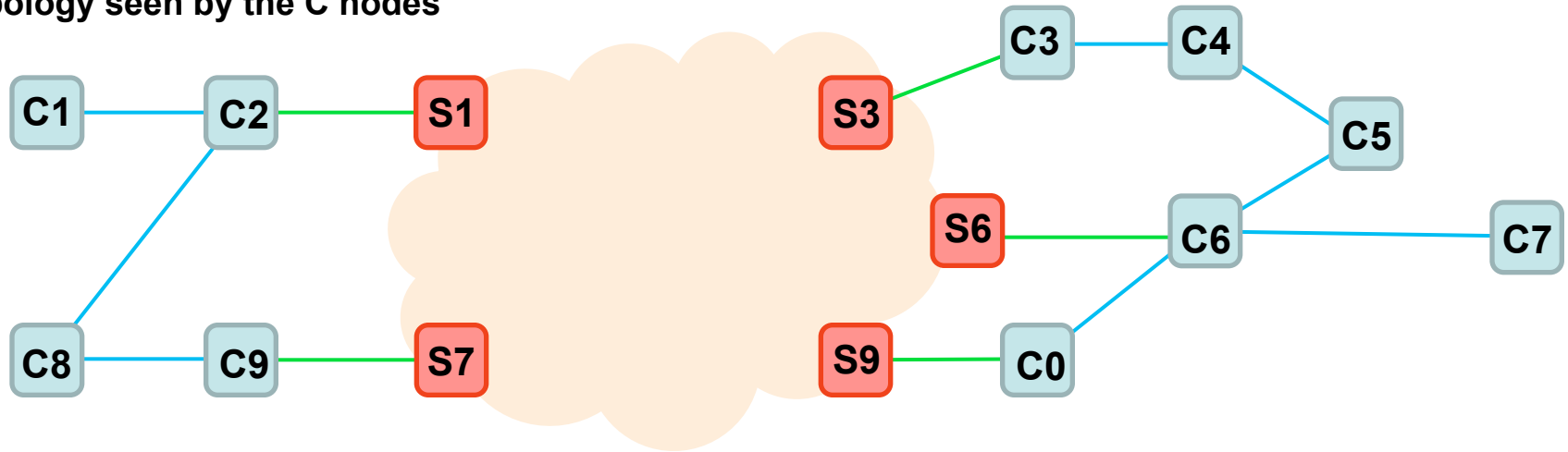
- Consolidate agreed use cases
- Discard not relevant ones
- Keep alignment with draft-farrel-interconnected-te-info

Computation model #1

Real Topology



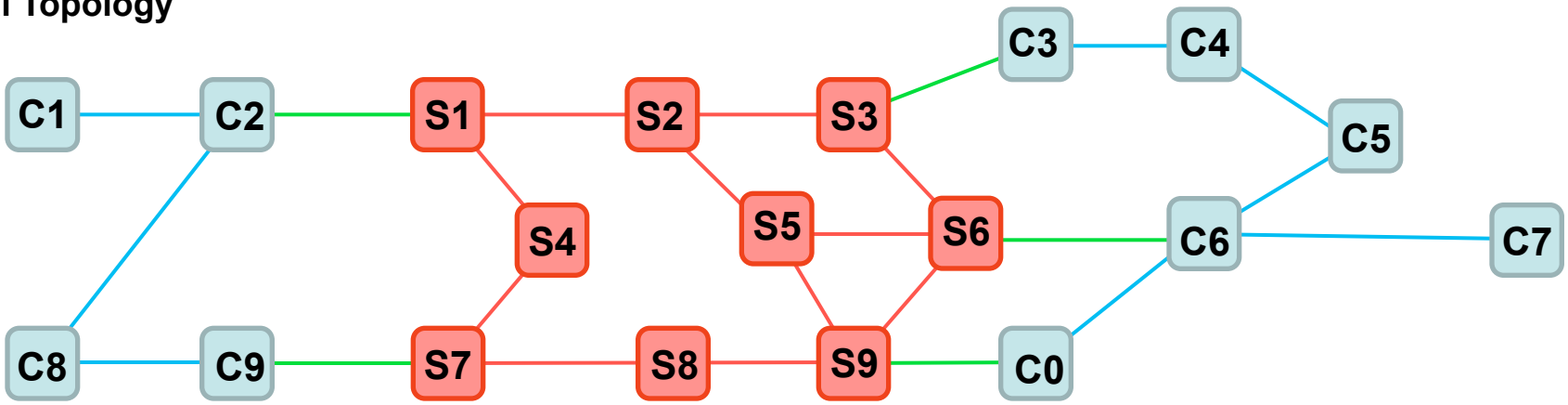
Topology seen by the C nodes



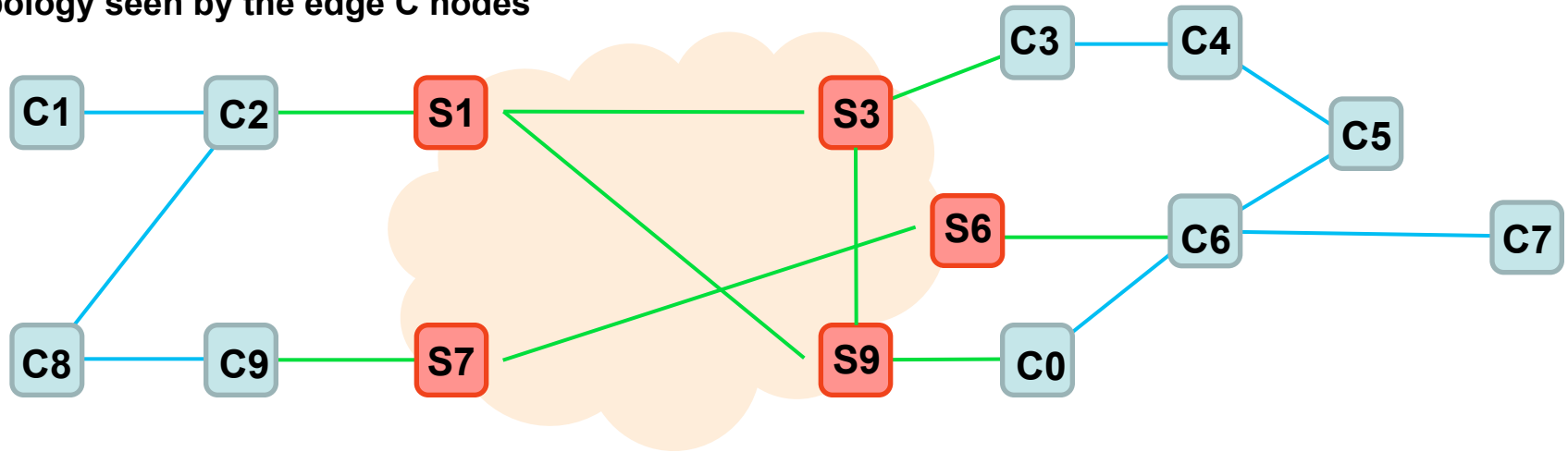
Service between C2 and C3 needs to be created. The operator decides that C2-S1 and S3-C3 will be used. The server layer computes and creates the server domain LSP between S1 and S3 with constraints

Computation model #2

Real Topology



Topology seen by the edge C nodes

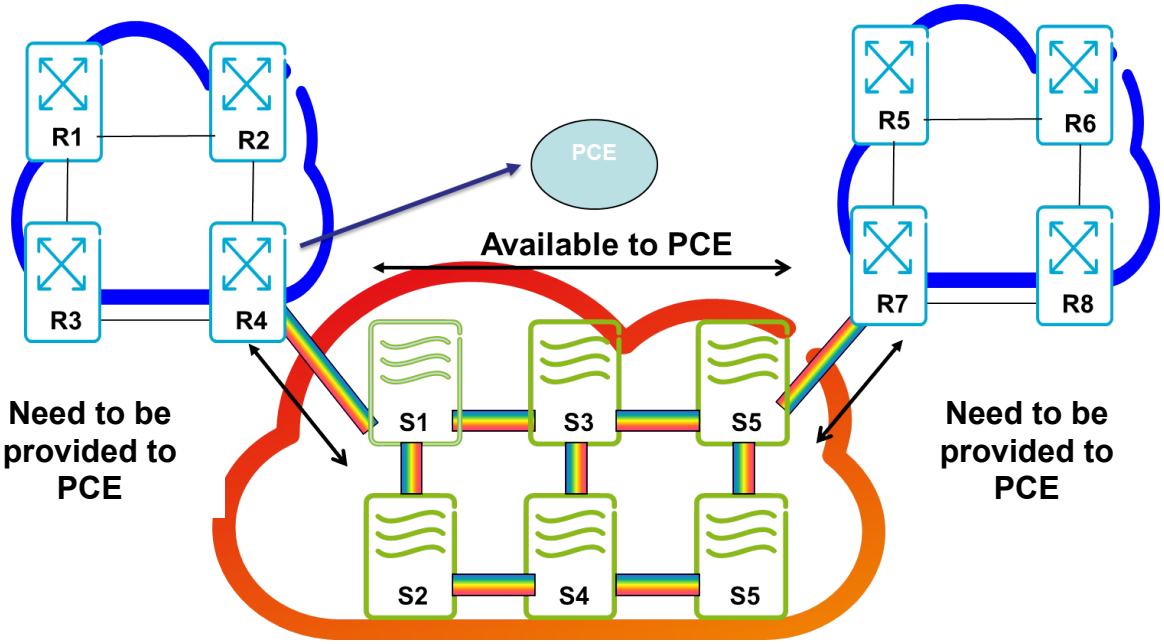


The abstract links (abstract topology) are computed a priori by the server domain (e.g. planning) and advertised with related TE info to the edge nodes. Edge nodes do a 3 hop (or more) path computation (e.g. service between C2 and C3 is computed along path C2-S1-S3-C3)

Appendix

- Colored overlay

Assumption: Path computation performed in the server layer



Feasibility: e.g. OSNR

Compatibility: e.g. modulation format

Availability: e.g. Lambda 1-3-7