Stitching in TIED: Collaborative Connectivity Establishment (Work in Progress)

Ted Faber, John Wroclawski, ISI

Problem

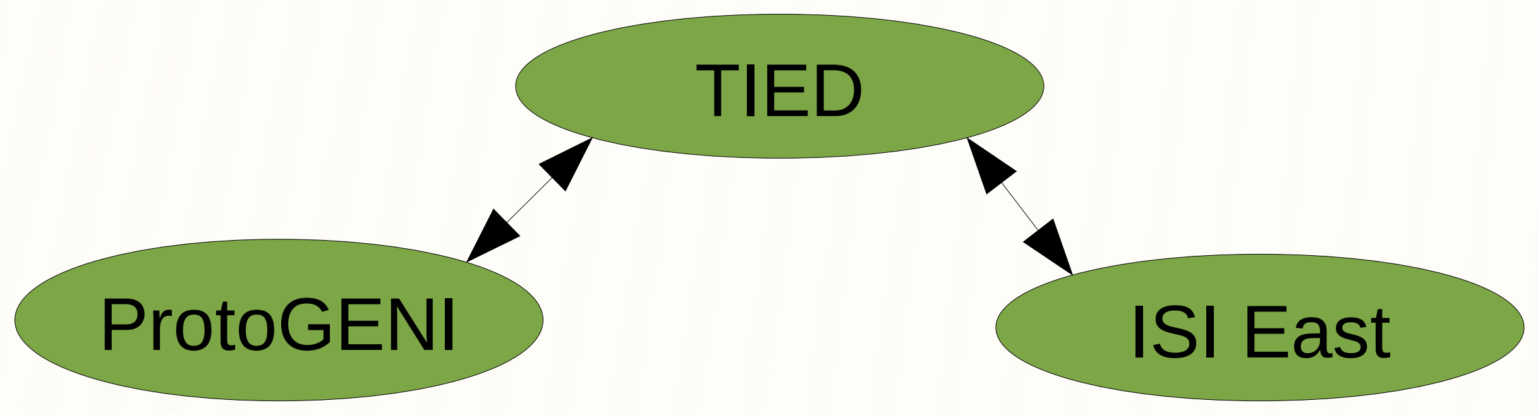
How does a federation system connect aggregates using external connectivity services, e.g., provisioned network services?

Approach

- TIED represents connectivity services with specialized *Connectivity Aggregates* that export simple interfaces
- Slice Manager:
  - Determines Aggregate connectivity preferences and constraints
  - Constructs a realizable negotiation path among Aggregates (including Connectivity Aggregates)
  - Acts as synchronization and data exchange point for Aggregates
- Aggregates negotiate through the Slice Manager to create links

Workflow

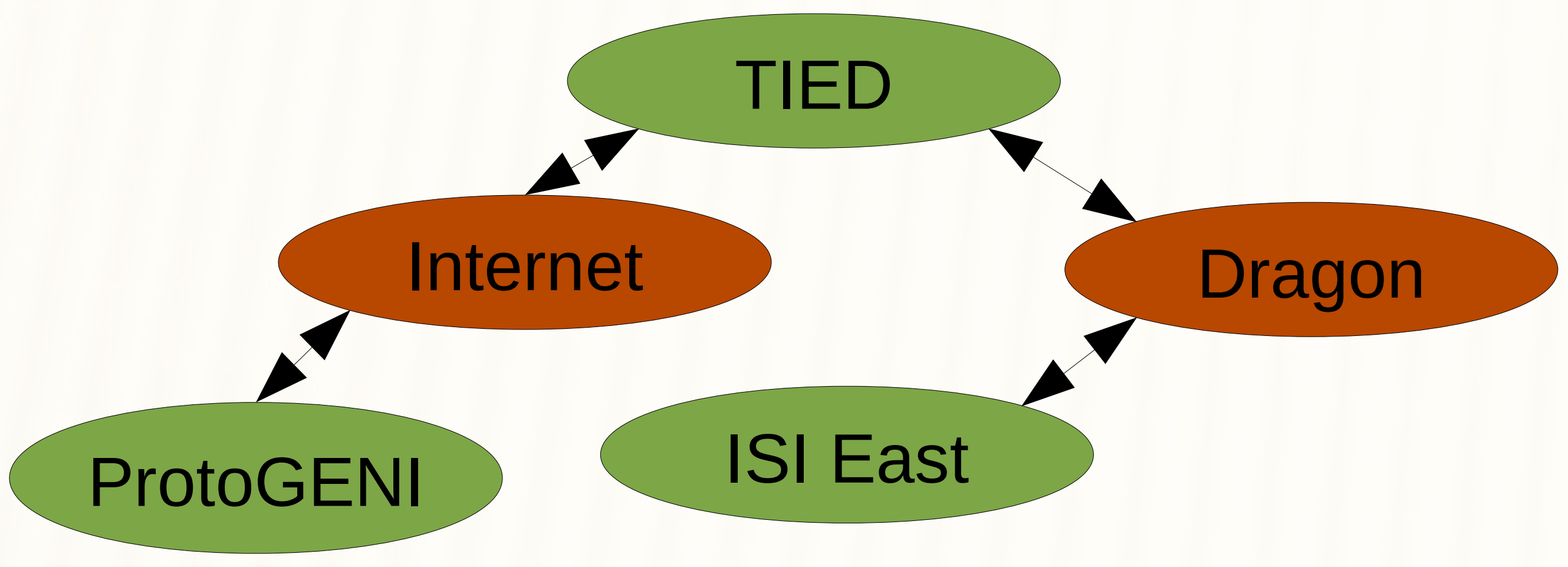
Slice Manager plans/analyzes Slice topology



Slice Manager gathers Aggregate connectivity preferences/constraints

```
Deter: Dragon
        needs label
        supplies label constraint
              Internet
        needs portal name
        supplies portal name
ISI East: Dragon
        needs label
        supplies label constraint
ProtoGENI: Internet,
        needs portal name
        supplies portal name
```

Slice Manager constructs topology with Connectivity Aggregates

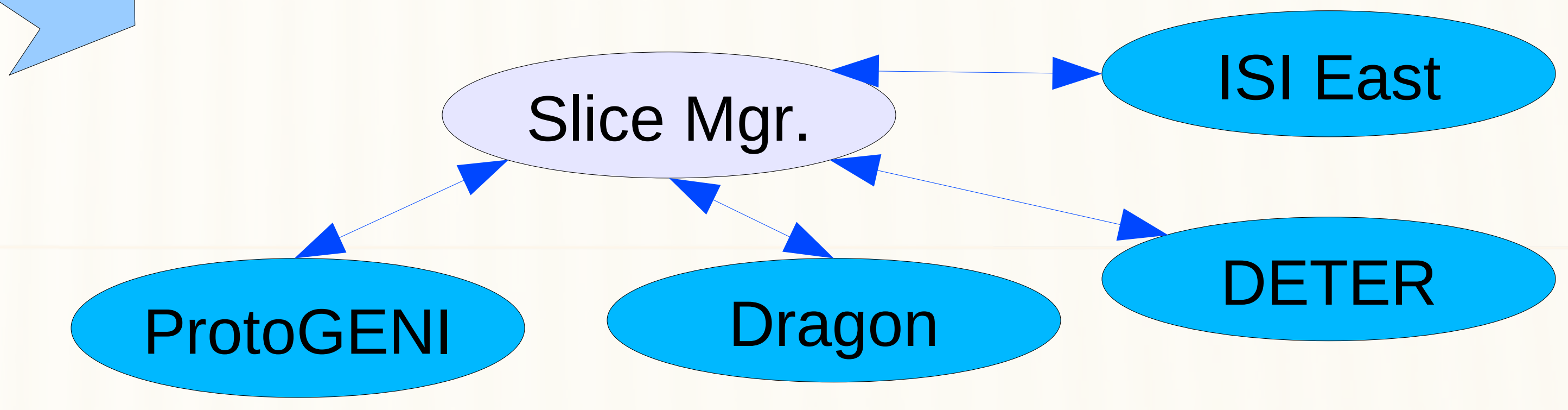


Slice Manager gets requirements constraints from Connectivity Aggregates

```
Dragon:
   takes label constraint
   supplies label
Internet:
```

Slice Manager confirms feasibility, assigns input/output keys, and starts creation. (Similar to Orca scheduling)

Aggregates exchange parameters through the Slice Manager during setup



Key Properties

Slice Manager sets the stage for feasible negotiations; aggregates negotiate
Avoids new communication requirements: Not all aggregates can communicate
Minimally constrains the parallelism of concurrent Sliver creations
The synchronized store can be scalably implemented, e.g. DHT
Initial implementation supported by current TIED plug-ins

GEC 7
16 March 2010