

Jonathon Duerig

Transactions and Update: A ProtoGENI Perspective

Why Transactions? (Create)

- Allocating resources is cheap
 - Failure is common
- Provisioning them is expensive
 - Failure is rarer

Why Transactions? (Update)

- Update implies
 - Existing Slivers
 - Allocation Change
 - Risk of Loss
- Transactions mitigate risk

Transactional Allocation

- Begin
 - “Guarantee”
 - Short-term Expiration
 - Slivers unchanged
- Commit
 - Slivers are updated
- Abort
 - No loss of sliver state

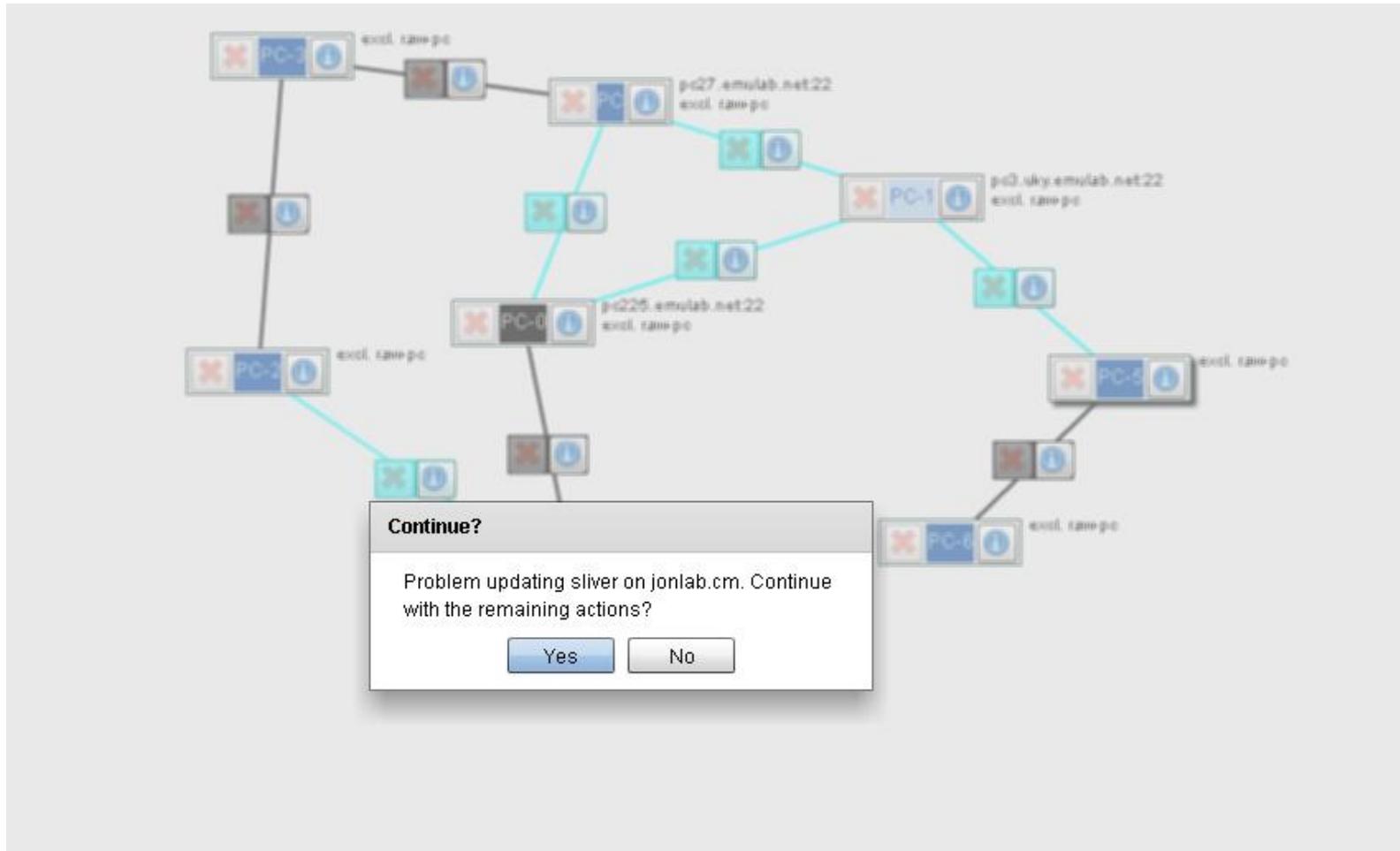
User Perspective (Update)

The screenshot displays the 'jon2' application interface. On the left, a sidebar contains a 'View' dropdown, an 'Import' button, and a list of three hosts: 'jonlab.cm', 'utahemulab.cm', and 'ukgeni.cm'. Each host entry includes a 'Get Sliver Status: Ready' indicator. The main area shows a network diagram on a light green background. Three PC nodes are visible, each with a red 'X' icon, a blue 'PC' label, and an information icon. The nodes are labeled as follows:

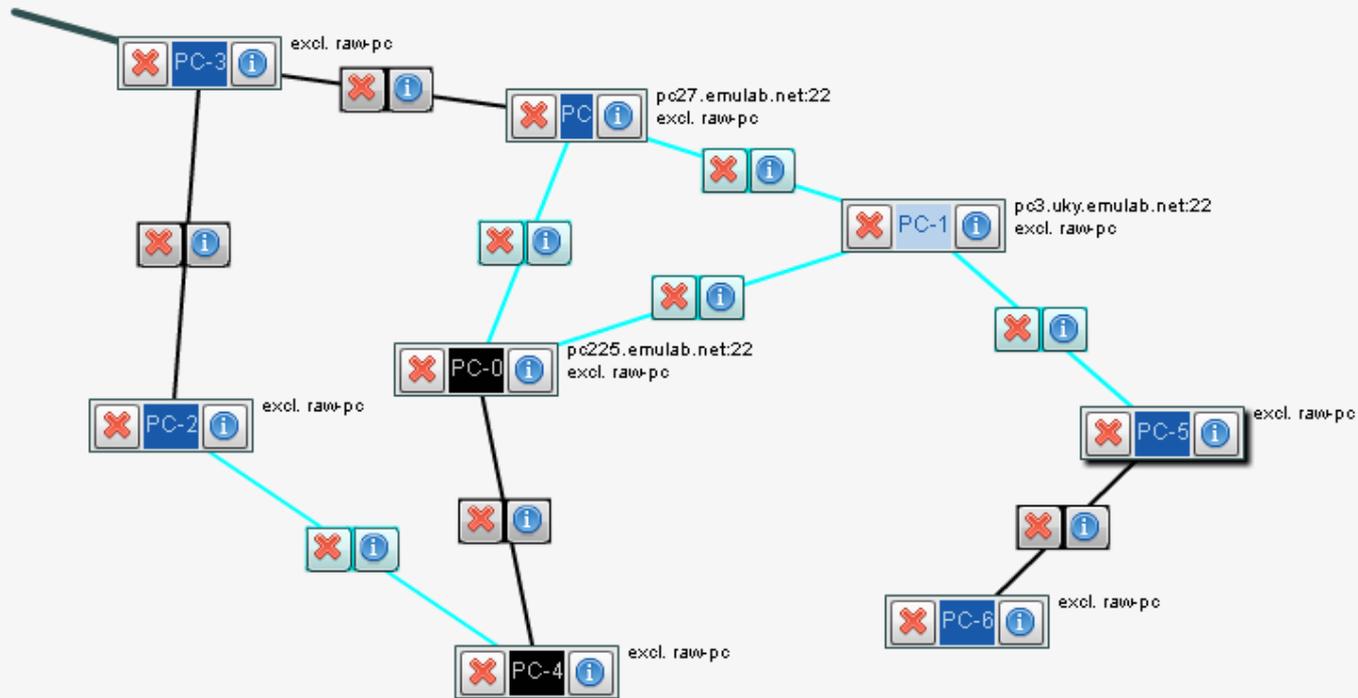
- Top node: pc27.emulab.net:22, excl. raw-pc
- Bottom node: pc225.emulab.net:22, excl. raw-pc
- Right node: pc3.uky.emulab.net:22, excl. raw-pc

Connections between the nodes are shown as blue lines. There are also several unlabelled nodes with red 'X' and information icons connected to the main nodes.

User Perspective (Problem)



User Perspective (Rollback)



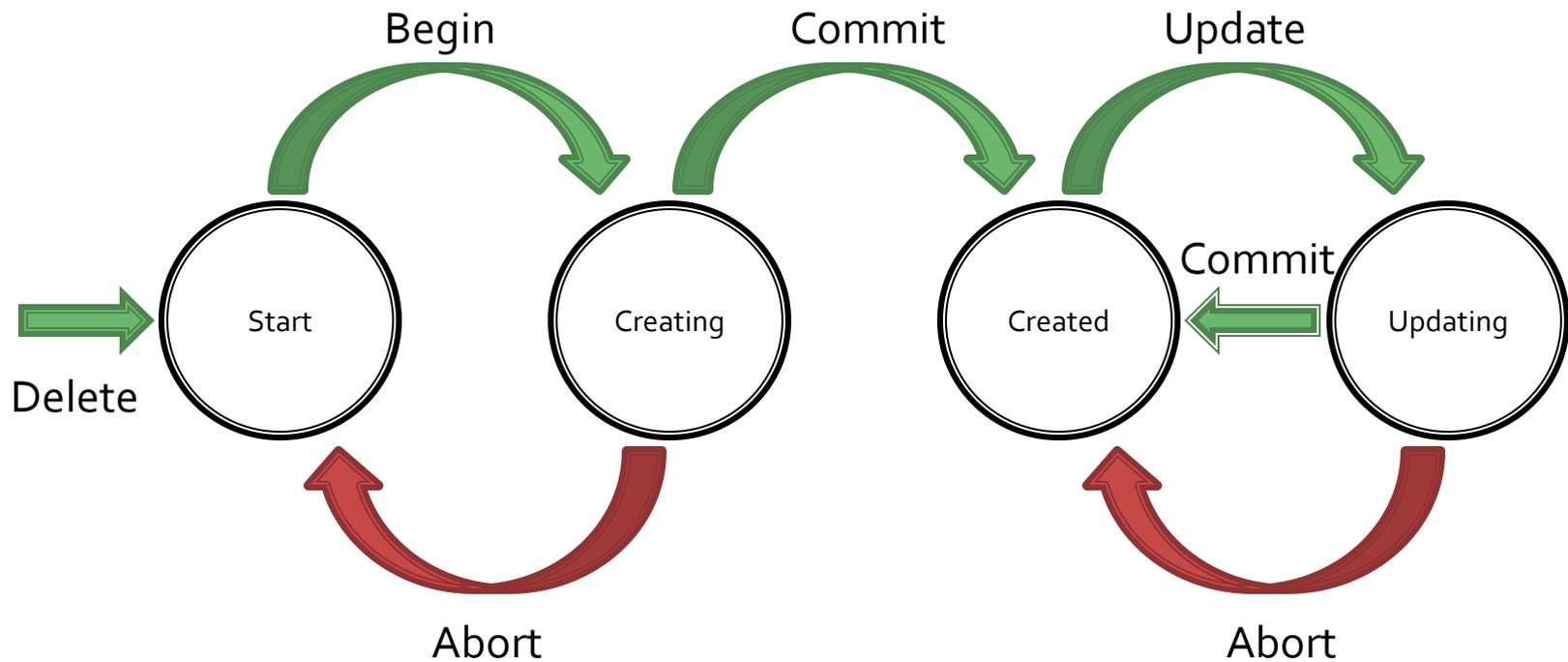
API Goals

- Allow user tools to enforce consistency
 - All or nothing?
 - Take what you get?
 - Change plans?
- Depends on the tool and user

Operation vs. Allocation

- Have I allocated resources at this AM?
 - Allocation state
 - Synchronous
 - Whole aggregate
- What is going on with my nodes?
 - Ground truth about resources
 - Asynchronous
 - Per-resource

API State Machine



Conclusion

- Support Commit/Rollback
 - Especially when updating
 - Need not be tickets
- Whole Aggregate
 - Allocation State Machine
- Per-Resource
 - Operational State Machine
 - Guarantees