TODOs for PyGNS3

elsholz

February 22, 2019

1 TODOs (Unordered)

- `Node.execute(self, command: str): {result: int, data: str}`
  Method for accessing a `Node`'s command line and execute a command on it. (3)

- `Node.plug_in(self, link: Link, port: int): None`
  Plug in one end of a `Link` into the `Node`'s port. (2)

- `Link.unplug(self, node): None`
  Unplug only one end of the link, which is plugged into the the node. (2)

- `Link.destroy(self): None`
  Unplug the `Link` on both ends. Acts like destructor and is invoked when `del Link` is called. (1)

- `Link.reconnect(self, end=None): None`
  Reverts the `Link` to the previous state, where both ends (or only the one given in `end`) were plugged in. (2)

- `Node.link_to(self, other): Link`
  Calls `Link.create(self, other)` and returns result. (1)

- `Link.create(node_a, node_b): Link`
  Creates a `Link` object based off two `Nodes`. (1)
  Optional: Overload the constructor thus that it can differ between direct and indirect construction.

- Refactor the code in order to increase readability. Focus lays on splitting (clusters of) classes between files.
  Create `Refactoring` branch for this purpose. (2)