HITACHI
Inspire the Next

Flow linter development

Kunihiko Toumura

Current status of Flow Linter development

- Repository: https://github.com/node-red/nrlint
- Implementation of CLI (alpha) version: completed (PR #3)
  - Extensible rule using npm module
    - `nrlint-plugin-core`:
      - **flowsize**: check number of nodes in single flow (tab)
      - **no-func-name**: check existence of name of function node
      - **http-in-resp**: check existence of corresponding http-response node for http-in node
      - **loop**: check existence of possible infinite loop
    - `nrlint-plugin-func-style-eslint`:
      - func-style-eslint: check code style in function nodes
  - They also can be used as plug-ins for Node-RED core, using current node plug-in mechanism, but it might be re-designed when new plug-in mechanism is designed and implemented.

Next steps of Flow Linter Development

- Design a generalized plug-in mechanism
  - There are also other use cases besides nrlint:
    - Embed node-gen to Node-RED to generate a node using Editor
    - Node discovery
      - Demonstrated in Web of Things Working Group
    - Runtime extension
      - e.g. distributed Node-RED using pluggable message routing API
    - Alternative context storage (e.g. using Redis)
  - How should we divide up the design process among us?

- Add more rules and extend Flow Manipulation API
  - Collecting ideas of rules
  - Add APIs to implement complex rules

- Documentation
  - Based on design note(https://github.com/node-red/designs/pull/1), write manuals for users/developers