

# The Node-RED v2.0 Core Nodes

## COMMON

- inject** - Injects a message into a flow either manually or at regular intervals.
- debug** - Displays selected message properties in the debug sidebar tab and optionally the runtime log.
- complete** - Trigger a flow when another node completes its handling of a message.
- catch** - Catch errors thrown by nodes on the same tab.
- status** - Report status messages from other nodes on the same tab.
- link in** - Create virtual wires between flows.
- link out** - Create virtual wires between flows.
- comment** - A node you can use to add comments to your flows.

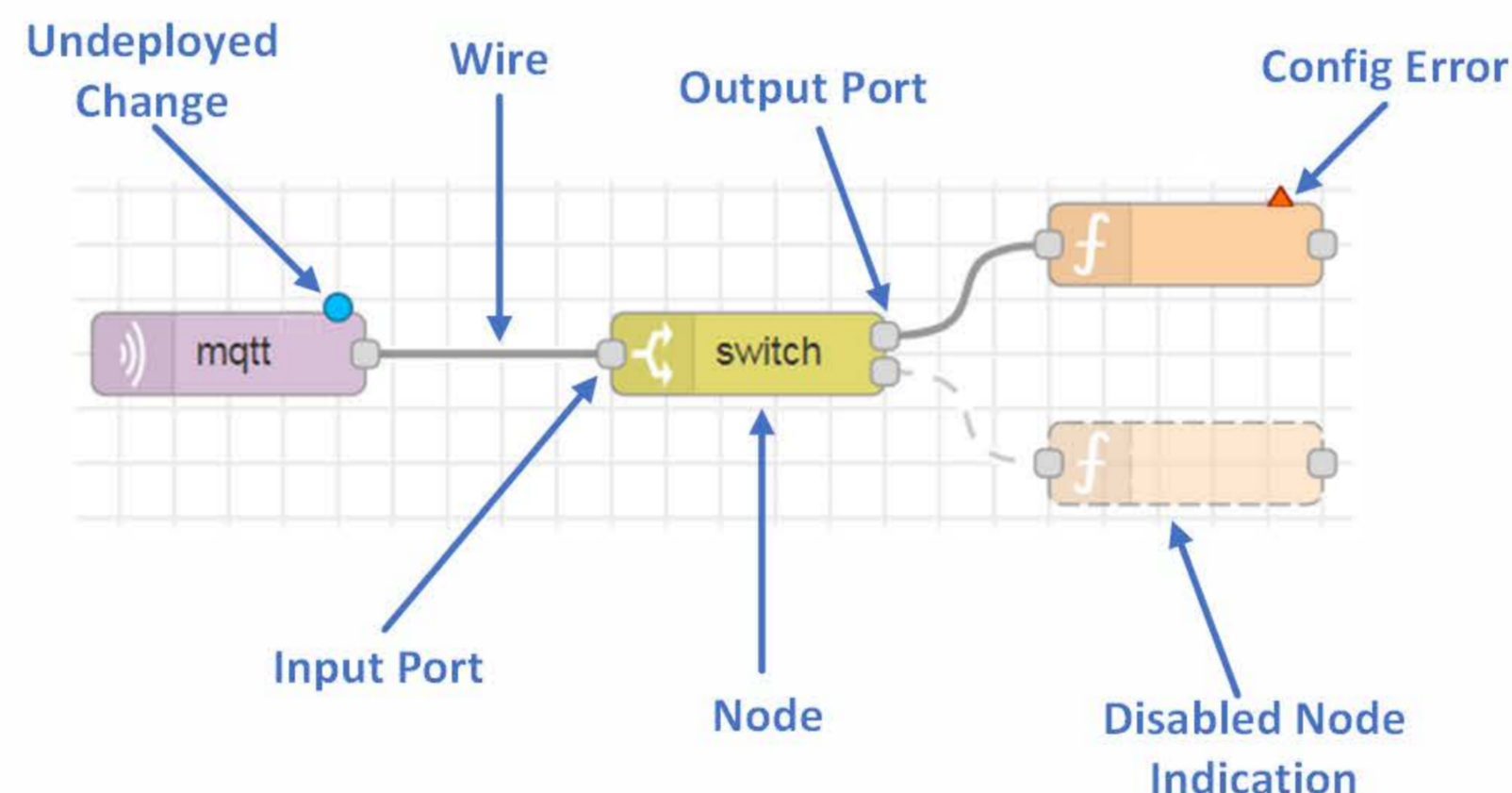
## FUNCTION

- function** - A JavaScript function block to run against the messages being received by the node.
- switch** - Route messages based on their property values or sequence position.
- change** - Set, change, delete or move properties of a message, flow context or global context.
- range** - Maps a numeric value to a different range.
- template** - Sets a property based on the provided template.
- delay** - Delays each message passing through the node or limits the rate at which they can pass.
- trigger** - When triggered, can send a message, and then optionally a second message, unless extended or reset.
- exec** - Runs a system command and returns its output.
- filter** - Report by Exception node - only passes on data if the payload has changed.

## SEQUENCE

- split** - Splits a message into a sequence of messages.
- join** - Joins sequences of messages into a single message.
- sort** - A function that sorts message property or a sequence of messages.
- batch** - Creates sequences of messages based on various rules.

## Flow Terminology



## PARSER

- csv** - Converts between a CSV formatted string and its JavaScript object representation, in either direction.
- html** - Extracts elements from an html document held in msg.payload using a CSS selector.
- json** - Converts between a JSON string and its JavaScript object representation, in either direction.
- xml** - Converts between an XML string and its JavaScript object representation, in either direction.
- yaml** - Converts between a YAML string and its JavaScript object representation, in either direction.

## NETWORK

- mqtt in** - Connects to a MQTT broker and subscribes to messages from the specified topic.
- mqtt out** - Connects to a MQTT broker and publishes messages.
- http in** - Creates an HTTP end-point for creating web services.
- http response** - Sends responses back to requests received from an HTTP Input node.
- http request** - Sends HTTP requests and returns the response.
- websocket in** - The socket can be configured to expect a properly formed JSON string, in which case it will parse the JSON and send on the resulting object as the entire message.
- websocket out** - The socket can be configured to encode the entire msg object as a JSON string and send that over the WebSocket.
- tcp in** - Provides a choice of TCP inputs. Can either connect to a remote TCP port, or accept incoming connections.
- tcp out** - Provides a choice of TCP outputs. Can either connect to a remote TCP port, accept incoming connections, or reply to messages received from a TCP In node.
- tcp request** - A simple TCP request node - sends the msg.payload to a server tcp port and expects a response.
- udp in** - A UDP input node, that produces a msg.payload containing a Buffer, string, or base64 encoded string. Supports multicast.
- udp out** - This node sends msg.payload to the designated UDP host and port. Supports multicast.

## STORAGE

- file** - Writes msg.payload to a file, either adding to the end or replacing the existing content.
- file in** - Reads the contents of a file as either a string or binary buffer.
- watch** - Watches a directory or file for changes.