

9.4 Node Red

Introduction to Node-RED

Node-RED is a flow-based programming tool built on Node.js that enables visual development of IoT applications and automation systems. Originally created by IBM for wiring together hardware devices, APIs, and online services, it provides a browser-based editor where developers create applications by connecting nodes together to form processing flows. Unlike Blynk's mobile-focused approach, Node-RED operates as a local server that can be deployed on various devices from single-board computers to cloud instances.

The platform uses a modular architecture where each node represents a specific function, such as reading sensors, processing data, or triggering actions. These nodes are connected together in sequences called flows, which define how data moves through the system and gets transformed along the way. This visual programming approach makes complex IoT logic more accessible to developers who may not have extensive coding experience.

Architecture and Deployment

Node-RED runs as a Node.js application, typically operating as a local web server on a designated port (default 1880). The runtime engine executes the flows, while the built-in web server provides both the visual editor and HTTP endpoints for external access. This architecture allows Node-RED to function as both a development environment and a runtime engine simultaneously.

A key advantage of Node-RED is its flexible deployment options. It can run directly on edge devices like Raspberry Pi for local processing, on servers for centralized control, or in cloud environments. The platform includes a built-in flow repository for saving and sharing configurations, and projects can be easily backed up through simple JSON file exports. This flexibility makes it suitable for everything from simple home automation to complex industrial IoT systems.

Core Concepts and Data Handling

The fundamental building blocks in Node-RED are nodes, which are categorized into three types: input nodes that inject data or receive external messages, processing nodes that transform or route data, and output nodes that send data to external systems or devices. Nodes are connected by wires that carry message objects containing payload data and optional properties.

Node-RED handles data as JSON message objects that flow between connected nodes. Each message contains a payload property that holds the primary data, along with optional topic and metadata properties. The platform includes numerous built-in nodes for common IoT protocols

including MQTT, HTTP, WebSockets, and serial communication. Additional functionality can be added through the extensive package manager, which provides thousands of community-contributed nodes for specialized hardware and services.

Integration Capabilities and Use Cases

Node-RED excels at integrating diverse systems and protocols within a single visual environment. It can connect to databases, web services, messaging systems, and hardware devices simultaneously, making it effective for building complex IoT gateways. The platform includes built-in support for function nodes where developers can write custom JavaScript code for specialized processing logic that isn't available in pre-built nodes.

Common IoT applications for Node-RED include data aggregation from multiple sources, protocol translation between different IoT devices, automation rule execution, and data visualization through dashboard nodes. The platform's ability to handle both real-time processing and scheduled tasks makes it suitable for monitoring systems, smart home automation, industrial control systems, and data processing pipelines. Unlike cloud-dependent platforms, Node-RED can operate completely locally, providing greater control over data privacy and system reliability.

Revision #1

Created 2025-11-09 14:33:38 UTC by Digilab UI

Updated 2025-11-09 14:34:31 UTC by Digilab UI