

## BIFROST: GATEWAY MODULAR DE COMUNICAÇÃO ENTRE DISPOSITIVOS IOT

Kerlon Kauã Ribeiro, Thales Gabriel Fernandes Diniz Martins, Leonardo Luís Röpke

### RESUMO

A expansão da Internet das Coisas (IoT) intensificou o uso de protocolos heterogêneos, criando barreiras de interoperabilidade que limitam a integração entre sistemas. Este trabalho propõe a Bifrost, um gateway modular e escalável, implementado sobre Raspberry Pi, que conecta ESP-NOW, MQTT e LoRa. Seu design prioriza flexibilidade e eficiência energética, padronizando mensagens em JSON e aplicando algoritmos de roteamento em Python para tradução e encaminhamento inteligentes. A metodologia envolveu testes de comunicação em três cenários: (i) ESP-NOW→ESP-NOW, (ii) ESP-NOW→MQTT e (iii) ESP-NOW→LoRa. Os resultados revelaram que a tradução ESP-NOW→MQTT apresentou melhor equilíbrio, com taxa de sucesso e latência. Os achados demonstram a viabilidade do gateway para integrar dispositivos heterogêneos, contribuindo para aplicações críticas em monitoramento ambiental, agricultura de precisão e automação industrial. Como perspectiva, destaca-se a possibilidade de ampliar a solução para protocolos adicionais, incorporar mecanismos avançados de segurança e otimizar o código desenvolvido, explorando melhorias de software e algoritmos que aumentem a eficiência e a confiabilidade do sistema.

Palavras-chave: Gateway IoT. Protocolos. Interoperabilidade. Computação de Borda.

### 1 INTRODUÇÃO

A Internet das Coisas (IoT) tem impulsionado a transformação digital em setores como agricultura, indústria e cidades inteligentes, permitindo a integração de dispositivos heterogêneos em escala global. Estimativas indicam que até 2030 haverá mais de 25 bilhões de dispositivos conectados, movimentando um mercado superior a US\$1 trilhão (Statista, 2023). Esse crescimento intensifica a demanda por soluções de comunicação capazes de superar a fragmentação de protocolos.

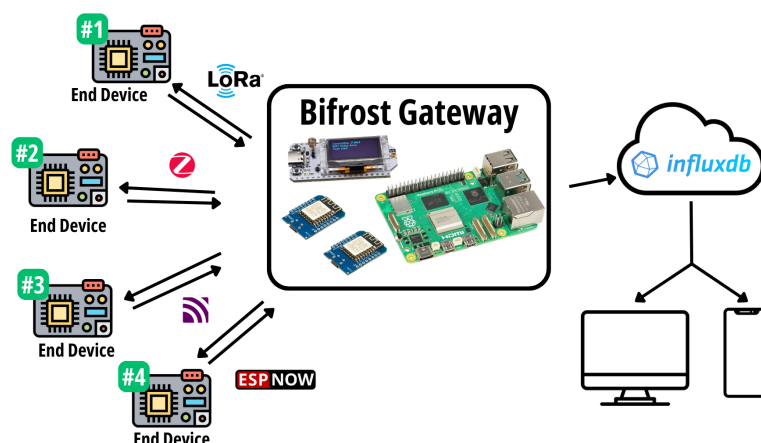
A diversidade de tecnologias sem fio, como ESP-NOW, MQTT, Zigbee e LoRa, resulta em redes isoladas, conhecidas como “ilhas de conectividade”. Essa heterogeneidade compromete a interoperabilidade e dificulta a escalabilidade dos sistemas, sendo apontada como um dos principais desafios atuais da IoT, ao lado da segurança e da privacidade (Fabri, 2025; Hmissi; Ouni, 2024; Kawthekar, 2025).

Nesse contexto, gateways desempenham papel estratégico ao traduzir protocolos, uniformizar mensagens e permitir a comunicação entre dispositivos distintos. Busca-se assegurar o gerenciamento otimizado do fluxo de dados por meio de tradução, filtragem e roteamento inteligentes, promovendo uma comunicação uniforme e escalável em redes heterogêneas. Contudo, soluções comerciais geralmente apresentam custos elevados, são proprietárias ou restritas a um número reduzido de tecnologias.

A principal contribuição deste trabalho é, portanto, o desenvolvimento da Bifrost, um gateway modular de baixo custo, implementado sobre Raspberry Pi, capaz de integrar ESP-NOW, MQTT e LoRa. A solução proposta permite a criação

de ambientes IoT mais integrados, resilientes e flexíveis. Ao atuar como uma ponte eficiente entre tecnologias distintas, o projeto tem impacto direto em áreas que demandam escalabilidade, como agricultura de precisão, monitoramento ambiental e automação residencial e industrial.

Figura 1 - Representação da Arquitetura do Sistema



Fonte: Autores (2025).

## 2 FUNDAMENTAÇÃO TEÓRICA

Para compreender a arquitetura e as decisões de projeto, é essencial revisar os conceitos tecnológicos fundamentais que o sustentam. Esta seção detalha o paradigma da computação de borda e os principais protocolos de comunicação selecionados para integração.

### 2.1 Computação de Borda (Edge Computing)

A computação de borda é um paradigma de computação distribuída que aproxima o processamento e o armazenamento de dados das fontes onde são gerados. Conforme definido por Hamdan et al. (2020), trata-se de uma arquitetura onde o armazenamento de dados, os serviços e as aplicações são movidos de nós centralizados (nuvem) para mais perto do usuário final. Em aplicações de IoT, essa abordagem oferece vantagens cruciais como a redução da latência e do tráfego de dados, além do aumento da autonomia do sistema, que pode operar mesmo com conectividade intermitente com a nuvem. O gateway Bifrost é uma materialização deste conceito, atuando como um nó de processamento local inteligente.

### 2.2 Protocolos de Comunicação

O ecossistema IoT é caracterizado por uma ampla gama de protocolos, cada qual otimizado para requisitos específicos como alcance, consumo de energia,

confiabilidade e custo. Essa diversidade, embora útil em contextos particulares, acentua o problema da interoperabilidade (Hmissi; Ouni, 2024).

- **MQTT (Message Queuing Telemetry Transport):** protocolo leve, baseado no modelo *publish/subscribe* e transportado via TCP, adequado a dispositivos com recursos limitados. Estudos apontam seu bom equilíbrio entre eficiência energética e segurança, mesmo com criptografia TLS habilitada (Vieira et al., 2024).
- **ESP-NOW:** protocolo proprietário da Espressif que permite a comunicação direta entre dispositivos sem a necessidade de ponto de acesso Wi-Fi. É indicado para aplicações que exigem baixa latência e respostas rápidas, como ambientes industriais (Santos et al., 2022).
- **LoRa (Long Range):** tecnologia de comunicação sem fio de longo alcance e baixo consumo de energia, capaz de atingir distâncias de até dez quilômetros em cenários favoráveis. É apropriada para monitoramento remoto em áreas rurais e urbanas (Branch; Weinstock, 2024).

Apesar da existência de outros protocolos relevantes — como Zigbee, Bluetooth Low Energy e CoAP — a escolha por integrar ESP-NOW, MQTT e LoRa no gateway Bifrost justifica-se pela relevância, usabilidade e prevalência em aplicações de automação. Essa seleção garante um equilíbrio entre desempenho, eficiência energética e escalabilidade, aspectos essenciais para aplicações em IoT.

### 3 METODOLOGIA

O desenvolvimento do gateway Bifrost seguiu uma abordagem experimental e iterativa, com foco na integração progressiva de protocolos em uma arquitetura modular. A seção está organizada em três etapas: Arquitetura do sistema, Interoperabilidade e gerenciamento de dados e Procedimentos de validação e teste.

#### 3.1 Arquitetura do Sistema

A arquitetura foi estruturada de forma modular, tendo como núcleo computacional um Raspberry Pi 4B (4 GB RAM), responsável pela execução do algoritmo de roteamento. Para a comunicação sem fio, foram empregados microcontroladores dedicados a cada tecnologia: um ESP Wemos D1 Mini para ESP-NOW e um ESP LoRa 32 Heltec V2 para LoRa. A comunicação via MQTT foi gerenciada por um broker Mosquitto, executado em contêiner Docker no Raspberry Pi. As interfaces UART foram utilizadas para a comunicação entre o núcleo e os módulos, garantindo escalabilidade e independência entre protocolos.

#### 3.2 Interoperabilidade e Gerenciamento de Dados

Todas as mensagens do sistema foram padronizadas no formato JSON unificado, contendo campos de origem (*src*), destino (*dst*), tipo (*type*) e dados da mensagem (*payload*). Um algoritmo em Python v3.11.2 foi implementado para

interpretar, traduzir e rotear as mensagens entre os módulos. Para registro e análise, utilizou-se o banco de dados InfluxDB, que permitiu monitoramento em tempo real e posterior consulta dos indicadores de desempenho.

### 3.3 Procedimentos de Validação e Teste

A validação experimental concentrou-se em medir latência e taxa de perda de pacotes. Para isso, os dispositivos emissores foram posicionados a aproximadamente 5 metros do gateway. O método de medição consistiu em um protocolo de ping-pong, no qual o tempo entre envio e resposta foi registrado como Round Trip Time (RTT). Um pacote foi considerado perdido quando a resposta excedeu 2 segundos. Foram avaliados três cenários de comunicação: (i) ESP-NOW→ESP-NOW, (ii) ESP-NOW→MQTT e (iii) ESP-NOW→LoRa. Cada cenário foi repetido em múltiplas execuções, totalizando 102.180 interações. Para análise quantitativa, foram calculadas as médias e percentuais de sucesso/perda em cada cenário.

## 4 APRESENTAÇÃO, ANÁLISE E DISCUSSÃO DOS RESULTADOS

Foram realizados 102.180 testes de comunicação, distribuídos entre três cenários experimentais. A Tabela 1 resume os principais resultados, considerando taxa de sucesso, taxa de perda e latência média (RTT).

Tabela 1 - Resultados dos Testes de Comunicação

Cenário	Pacotes enviados	Respostas recebidas	Taxa de sucesso	Taxa de perda	Latência média
ESP-NOW → ESP-NOW	64.050	44.279	69,1%	30,9%	667 ms
ESP-NOW → MQTT	14.430	12.268	85,0%	15,0%	523 ms
ESP-NOW → LoRa	23.700	12.586	53,1%	46,9%	1.160 ms

Fonte: Autores (2025).

A análise evidencia que o MQTT apresentou os melhores resultados, combinando baixa latência e menor perda de pacotes, o que confirma sua adequação para aplicações que exigem maior confiabilidade. O ESP-NOW puro obteve latência intermediária, mas com perdas significativas, possivelmente associadas a limitações do protocolo. O LoRa, por sua vez, apresentou o pior desempenho em termos de latência e taxa de sucesso, comportamento esperado devido às características de transmissão de longo alcance e baixa taxa de dados. Esses achados estão alinhados à literatura, que aponta o LoRa como mais suscetível a atrasos (Branch; Weinstock, 2024) e o MQTT como opção eficiente mesmo em cenários restritivos (Vieira et al., 2024). Assim, o gateway Bifrost demonstrou viabilidade para integrar protocolos complementares, atuando como solução prática para ambientes IoT heterogêneos.

---

## 5 CONCLUSÃO

O projeto Bifrost comprovou a viabilidade de um gateway modular e de baixo custo para enfrentar o desafio da interoperabilidade em ecossistemas IoT. A integração entre ESP-NOW, MQTT e LoRa demonstrou que a arquitetura é capaz de processar grandes volumes de dados com flexibilidade e escalabilidade, ainda que com limitações de latência e perda de pacotes em determinados cenários. Como trabalhos futuros, destaca-se a incorporação de novos protocolos, como Zigbee e LoRaWAN, o aprimoramento de mecanismos de segurança e a otimização do software e dos algoritmos já desenvolvidos, de modo a elevar a eficiência e a confiabilidade do sistema em aplicações críticas.

## 6 REFERÊNCIAS

BRANCH, P.; WEINSTOCK, P. Functional Programming for the Internet of Things: A Comparative Study of Implementation of a LoRa-MQTT Gateway Written in Elixir and C++. *Electronics*, v. 13, n. 17, p. 3427, 2024.

FABRI, E. V. et al. A Revolução da Internet das Coisas (IoT). *Revista Missioneira*, v. 27, n. 1, p. 173–183, 18 abr. 2025.

HAMDAN, S.; AYYASH, M.; ALMAJALI, S. Edge-Computing Architectures for Internet of Things Applications: A Survey. *Sensors*, v. 20, n. 22, p. 6441, 2020.

HMISSI, F.; OUNI, S. A Survey on Application Layer Protocols for IoT Networks. Disponível em: <<https://arxiv.org/abs/2405.15901>>. Acesso em: 20 set. 2025.

KAWTHEKAR, D. et al. A Comprehensive Survey on Internet of Things (IoT), Architecture, Protocols, and Applications. Disponível em: <<https://www.ijraset.com/research-paper/iot-architecture-protocols-and-applications>>. Acesso em: 20 set. 2025.

SANTOS, L. H. et al. Utilização dos protocolos ESP-NOW e MQTT para monitoramento e interação com sistemas industriais. *Anais do Simpósio Brasileiro de Sistemas Elétricos*, 2022.

STATISTA. *Internet of Things (IoT): number of connected devices worldwide from 2019 to 2030*. 2023. Disponível em: <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>. Acesso em: 21 set. 2025.

VIEIRA, E. et al. Análise de Desempenho e Eficiência Energética dos Protocolos MQTT e CoAP no contexto de IoT. p. 321–327, 16 set. 2024.