

Estudo sobre os modos de integração do MQTT na plataforma InterSCity

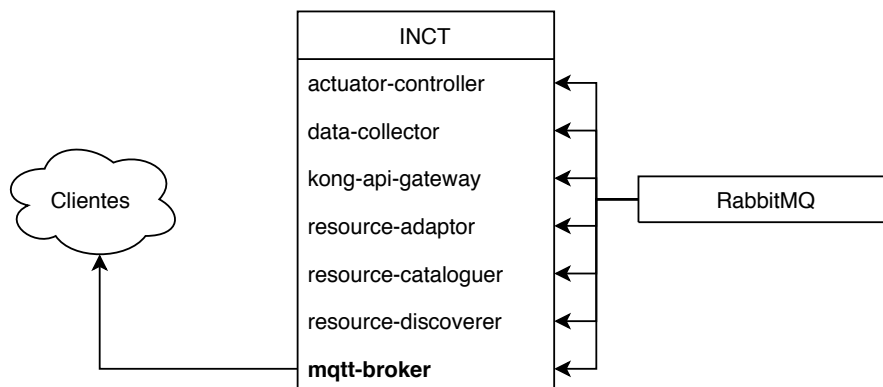
Bruno Carneiro da Cunha
<brunocarneirodacunha@usp.br>

26 de maio de 2020

1 Possíveis modos de integração

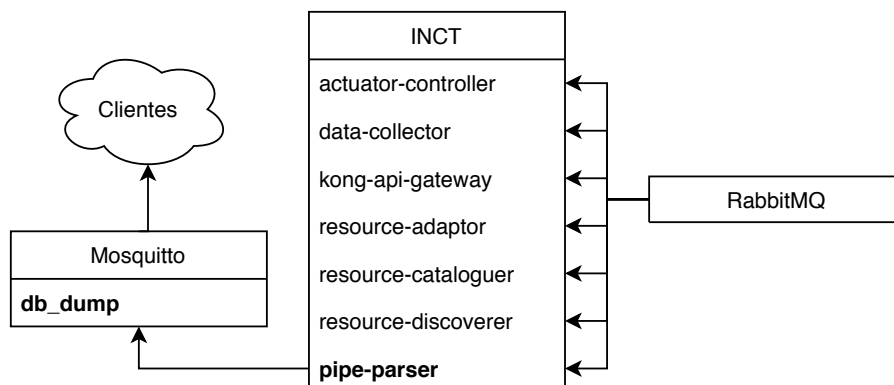
1.1 Implementação de *broker* como módulo

Seguindo a arquitetura da plataforma, o módulo MQTT seria um *broker* desenvolvido em Ruby, que executaria todas as funções de *broker*, e também cumpriria a função de fornecer os dados MQTT aos outros módulos da plataforma.



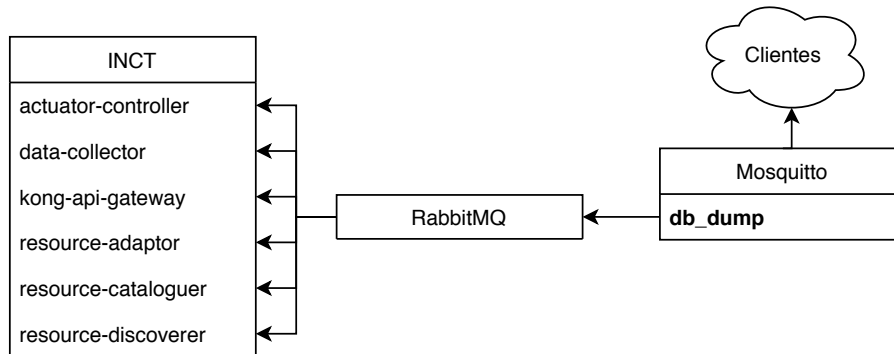
1.2 *Broker* MQTT local - comunicação via *pipe*

Todas as funções de servidor MQTT seriam cumpridas por uma instância do *Mosquitto*, que estaria instalado na mesma máquina da plataforma. Seria necessário então desenvolver um módulo adicional à plataforma que consumiria as informações através da leitura dos arquivos do banco de dados do *Mosquitto*.



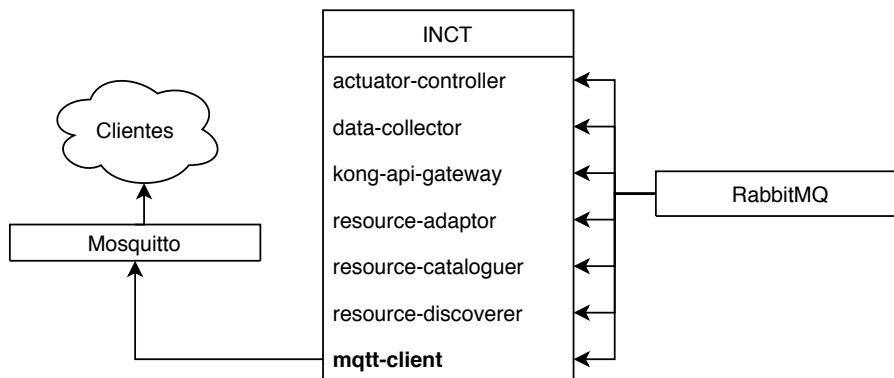
1.3 *Broker* MQTT local - comunicação via RabbitMQ

Esse seria um cenário mais otimizado, onde o próprio *Mosquitto* seria modificado para se comunicar diretamente com os módulos da plataforma como um cliente do servidor RabbitMQ que já existe.



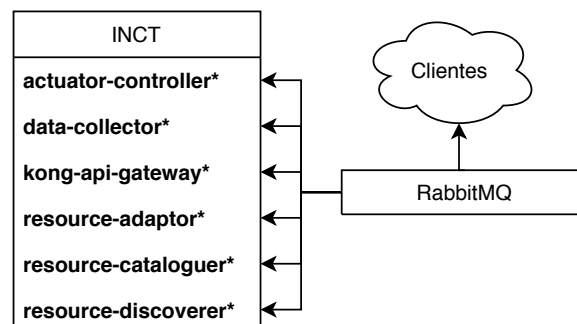
1.4 *Broker* MQTT remoto/local - comunicação via MQTT

Todas as funções de servidor MQTT seriam cumpridas por uma instância do *Mosquitto*, que poderia estar instalado em qualquer máquina. A plataforma teria um módulo adicional que seria um cliente MQTT. Esse assinaria todos os tópicos do servidor e assim receberia todos os dados.



1.5 RabbitMQ como *broker*

Nesse cenário, simplesmente ativaríamos o plugin MQTT do servidor RabbitMQ que já roda na plataforma. Os módulos poderiam então ser modificados para consumir também os dados MQTT.



2 Tabela de avaliação

#	Descrição	Vantagens	Desvantagens
1	Broker como módulo	- Alta customização	- Alta complexidade - Correção de vulnerabilidades seria da responsabilidade do mantenedor da plataforma
2	Broker local Pipe	- O consumo dos dados envolve somente ler um arquivo	- Dificuldade para manter o broker atualizado com as modificações - Atualização no broker pode quebrar o parse dos arquivos - Seria necessário ainda implementar um cliente MQTT na plataforma para escrever em tópicos caso necessário
3	Broker local RabbitMQ	- Eficiência	- Dificuldade para manter o broker atualizado com as modificações - Complexidade de adaptar um cliente AMQP à estrutura do Mosquitto
4	Broker remoto MQTT	- Simplicidade na implementação - Flexibilidade para usar qualquer servidor MQTT	
5	Plugin MQTT RabbitMQ	- Serviço já existe no INCT, só é necessário ativar o plugin - Interoperabilidade entre protocolos já existe no RabbitMQ	- Não sabemos se a performance do plugin é satisfatória - O plugin não permite conexões anônimas