# INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

**Área Departamental de Engenharia de Eletrónica e Telecomunicações e de Computadores**

**Licenciatura de Engenharia Informática e de Computadores**

**Unidade Curricular de Sistemas Distribuídos**

**2º semestre letivo 2017/2018**

## Modelo de Entrega de Dados para Sistemas Publish-Subscribe Baseado em Fog/Cloud

**Grupo G5**

**Autores: 42181 Nuno Veloso, 42798 Steven Brito, 42799 Daniela Gomes**

**Introdução:**

O modelo *Publish-Subscriber* usado em várias aplicações onde subscritores indicam o seu interesse num determinado tópico usando palavras chaves relacionadas e são notificados quando alguém publica novos dados sobre o referido tópico no canal. Porém o desenvolvimento de sistemas para suportar aplicações distribuídas usando este modelo em larga escala é um problema recorrente.

As arquiteturas mais usadas para a resolução do modelo *Pub/Sub* são *Peer to peer* (P2P) e *Broker Overlay*. Na arquitetura P2P um *subscriber* pode ser também um *publisher* e vice-versa. Isto permite que o sistema seja capaz de suportar uma elevada quantia de tópicos e conteúdos. Contudo, não é fiável devido à desistência de muitos participantes que pode tornar o sistema menos eficiente.

Na arquitetura *Broker Overlay* vários *pub/sub brokers* estão organizados numa *Broker Overlay Network* em que os participantes conectam-se a alguns brokers para publicar ou subscrever um tópico desejado. Embora esta arquitetura garanta tolerância a falhas e soluções economicamente viáveis não permite a adição dinâmica de brokers, tornando difícil a escalabilidade.

É com este propósito que o artigo *“A Fog/Cloud Based Data Delivery Model for Publish-Subscriber Systems”[]* propõe uma solução que consiste numa hierarquia de brokers e participantes para a entrega dos eventos. Para a coordenação entre as diversas componentes usam um conjunto de servidores *Zookeeper[]* de forma a aprimorar a escalabilidade e o desempenho destes sistemas.

#### Maio de 2018

# Síntese

## Modelo:

No artigo é descrita uma solução para o modelo *Publish-Subscriber*. Este modelo consiste em ter principalmente três componentes: Servidores coordenadores, brokers *Pub/Sub* e participantes.

Os servidores coordenadores consistem em servidores *ZooKeeper* (ZK) para coordenar os *brokers pub/sub* e os participantes. Usaram esta tecnologia por ser capaz de eleger um líder, detetar falhas, ter a associação de grupos e é possível gerir as configurações nos sistemas distribuídos.

Existem também servidores de processamento que realizam tarefas como mapeamento de participantes, deteção de novos tópicos, atribuição de *brokers* aos participantes designados, entre vários outros.

Os grupos de *brokers* são lançados em localizações geográficas de modo estratégico para reduzir a latência entre o envio das mensagens. Estes conectam-se aos servidores ZK para coordenarem-se entre si dinamicamente e tornar a coordenação mais eficaz num ambiente distribuído.

O servidor ZK usa uma hierarquia de nós chamada *ZNodes[]*. Cada *ZNode* pode armazenar uma quantia relativamente pequena de dados contem versões e *timestamps* para ler e alterar atomicamente os dados contidos.

## Funcionamento:

A figura 1 descreve a hierarquia do modelo znode proposta no artigo.

* Conectar participantes ao sistema.

Um participante começa por conectar-se a um servidor ZK, criando um znode que representa o pedido de mapeamento, contendo o seu endereço IP e o porto. O líder dos servidores de processamento atribui a tarefa de mapeamento a um worker com menos carga de trabalho. O worker selecionado procura pela localização do cliente e utiliza técnicas descritas em [….] para determinar os grupos de brokers (Fog/Cloud) mais próximos do participante. O participante ao estar inserido dentro do contexto de um grupo cria um znode para representar o pedido de correspondência com brokers Pub/Sub apropriados, contendo os tópicos que deseja subscrever/publicar.