Daniel Filipe Santos Pimenta

Mestrado Integrado em Engenharia Informática

Gestão Dinâmica de Micro-serviços na Cloud/Edge

Relatório intermédio para obtenção do Grau de Mestre em

Engenharia Informática

Orientadora: Maria Cecília Farias Lorga Gomes, Profª. Auxiliar, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa

Co-orientador: João Carlos Antunes Leitão, Prof. Auxiliar, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Fevereiro, 2019

Resumo

Observa-se, hoje em dia, um crescimento muito elevado da utilização de dispositivos no domínio da Internet of Things (IoT) e de dispositivos móveis, bem como do número de aplicações com consumo elevado de largura de banda (ex.: visualização de vídeos, a pedido). Tal implica que, num futuro próximo, não será viável suportar a quantidade de dados transferidos entre os dispositivos clientes ("*end devices*") e os centros de dados *cloud*, onde tipicamente são alojadas aplicações de acesso ubíquo. O problema do consequente aumento da latência percebido nas aplicações clientes, é ainda agravado no caso de aplicações "sensíveis à latência" *(latency sensitive)*, como sejam aplicações bastante interativas ou de tempo real/quase-real (ex.: carros autonómicos, jogos online, etc.). A localização deste tipo de aplicações na *cloud*, onde é grande a distância entre os clientes e a localização dos centros de dados, resulta em níveis inaceitáveis de Quality of Service (QoS) percebida pelos clientes, ou mesmo a impossibilidade de cumprir os requisitos funcionais das aplicações.

A computação na *edge* (*Edge computing*) surge como resposta aos problemas de latência referidos, ao usar recursos computacionais dos dispositivos na periferia da rede, que se situam mais próximo das aplicações cliente. É ainda possível realizar computações que filtrem os dados gerados na periferia, contribuindo para diminuir o volume de dados em trânsito, e que teriam de ser processado na *cloud*.

Comparativamente com os recursos presentes nos centros de dados *cloud*, os recursos dos nós na *edge* são, no entanto, de capacidade computacional bastante limitada. Isto implica que utilizar aplicações monolíticas (tipicamente de grandes dimensões) não é uma opção eficaz na computação na *edge*, quer pelo custo da sua migração/replicação, quer pela impossibilidade de alojar as aplicações nesses nós. O uso da arquitetura de micro-serviços procura solucionar este problema. As aplicações são compostas por múltiplos micro-serviços, cada um com pequena dimensão, oferecendo uma funcionalidade única, com interfaces bem definidas e que comunicam entre si através de mensagens, tornando-os independentes entre si. Desta forma, é possível realizar uma gestão mais eficaz dos recursos disponíveis nos nós periféricos.

O trabalho proposto baseia-se num sistema, já existente, de gestão automática de migração/replicação de micro-serviços, dos centros de dados *cloud* para a periferia e entre

componentes periféricos. O objetivo do trabalho é estender esse sistema tendo em conta um conjunto mais alargado de métricas (ex.: taxa de ocupação do CPU e *throughput*, para além da latência) que permita utilizar os recursos limitados de modo mais eficiente, e permitir uma QoS adequada às aplicações. Será ainda re-avaliada a definição da arquitetura tendo em vista acomodar a existência de nós computacionais heterogéneos no *continuum* entre a *cloud* e a periferia, bem como as características das aplicações em termos do número de micro-serviços possíveis de replicar/migrar.

Palavras-chave: *Cloud*, *Edge*, Micro-serviço, Gestão de recursos

Abstract

The dissertation must contain two versions of the abstract, one in the same language as the main text, another in a different language. The package assumes that the two languages under consideration are always Portuguese and English.

The package will sort the abstracts in the appropriate order. This means that the first abstract will be in the same language as the main text, followed by the abstract in the other language, and then followed by the main text. For example, if the dissertation is written in Portuguese, first will come the summary in Portuguese and then in English, followed by the main text in Portuguese. If the dissertation is written in English, first will come the summary in English and then in Portuguese, followed by the main text in English.

The abstract shoul not exceed one page and should answer the following questions:

- What's the problem?
- Why is it interesting?
- What's the solution?
- What follows from the solution?

Keywords: Keywords (in English) ...

Índice

Lista de Figuras

Lista de Tabelas

Listagens

Glossário

economia de escala	Economias de escala são os fatores que conduzem à redução do custo médio de produção de um determinado bem à medida que a quantidade produzida aumenta.
hardware	.
latência	Período de latência é a diferença de tempo entre o início de um evento e o momento em que os seus efeitos se tornam percetíveis.
middleware	.
on-premises	*On-premises software*, também conhecido como *shrink wrap*, é um modelo de distribuição de software que é instalado e operado numa infraestrutura computacional presente no local do cliente.
software	.

Siglas

AMQP	Advanced Message Queuing Protocol.
CDN	Content Delivery Network.
CoAP	Constrained Application Protocol.
DDoS	Distributed Denial of Service.
DDS	Data Distribution Service.
HTTP	Hyper Text Transfer Protocol.
HTTPS	Hyper Text Transfer Protocol Secure.
IaaS	Infrastructure as a Service.
IEEE	Institute of Electrical and Electronics Engineers.
IoT	Internet of Things.
MCC	Mobile Cloud Computing.
MEC	Mobile Edge Computing.
MQTT	Message Queuing Telemetry Transport.
NIST	National Institute of Standards and Technology.
P2P	Peer-to-Peer.

PaaS Platform as a Service.

QoE Quality of Experience.

QoS Quality of Service.

SaaS Software as a Service.

SLA Service Level Agreement.

SLOs Service Level Objectives.

VOD Video on Demand.

Capítulo 1

Introdução

TODO: introdução/resumo do capitulo

1.1 Contexto e Motivação

O paradigma de computação centralizado, denominado por computação cloud (*cloud computing*), tem sido o mais predominante, com maior interesse e maior investimento nas últimas duas décadas. Implementado inicialmente na prática pela Amazon, e anos mais tarde seguido por gigantes do mundo tecnológico como a Google, Microsoft, IBM, entre outros, a computação cloud permite fornecer poder computacional, armazenamento e largura de banda como um serviço. Teve como motivação principal a eficiência do processamento da informação que é obtida, devido à economia de escala, em grandes centros de computação. E teve sucesso entre as entidades individuais e empresariais por poderem beneficiar de uma infinidade aparente de recursos computacionais a um preço utilitário.

Mas, com a previsão do crescimento da utilização de dispositivos IoT (Ex.: sensores para *smart cities*, *smart homes*, dispositivos *wearables*) e o aumento da exploração de aplicações com consumo elevado de largura de banda (Ex.: Video on Demand (VOD), streaming de conteúdo vídeo, 4k TV), prevê-se que a rede fique demasiado congestionada devido à grande quantidade de dados transferidos entre os dispositivos *front-end* (Ex.: telemóveis, *tablets*, *desktops*, computadores portáteis, televisões, consolas de jogos) e os centros de dados cloud. Também a distância entre estes dispositivos e os centros de dados cloud impõe uma limitação na interatividade das aplicações (Ex.: realidade virtual e aumentada) devido à latência elevada entre os mesmos.

Os avanços tecnológicos recentes nos dispositivos periféricos (Ex.: *routers*, *gateways*,

switchs e estações base) indicam uma possível mudança gradual de paradigma de computação como tentativa de resolver esses problemas. O paradigma de computação distribuída usando os dispositivos periféricos, denominado por computação edge (*edge computing*), pretende solucionar os problemas relacionados com a latência e a transferência elevada de dados entre dispositivos *front-end* e centros de dados cloud. Isto é conseguido ao usar dispositivos na periferia para fazer a ligação entre os dispositivos *front-end* e os centros de dados cloud. Como consequência, os dispositivos *front-end* usados pelos utilizadores podem beneficiar de uma redução de latência devido à proximidade aos dispositivos periféricos. Permite também a filtragem e agregação dos dados nos dispositivos periféricos reduzindo a quantidade de dados transferidos para a cloud.

Dito isto, surge outra incompatibilidade. Os recursos reduzidos dos dispositivos periféricos complicam a migração de aplicações monolíticas para os dispositivos periféricos por serem demasiado grandes e complexos. Aqui é introduzida a arquitetura de micro-serviços. Esta arquitetura permite desenvolver software com base em múltiplos micro-serviços, de pequena dimensão e *interfaces* bem definidas, que comunicam entre si através de mensagens. Os benefícios desta arquitetura não se limitam apenas à periferia. Cada micro-serviço pode implementar uma funcionalidade especifica e isolada permitindo, assim, o desacoplamento aos outros micro-serviços. Com isto, é também possível reduzir o tempo de migração/replicação de cada micro-serviço entre a periferia e a cloud, ou entre componentes periféricos, o que permite uma gestão mais eficiente da carga nos dispositivos periféricos.

A gestão de recursos na periferia da rede é exatamente o problema essencial que esta dissertação pretende abordar. O objetivo é fazer decisões quanto à migração e/ou replicação de micro-serviços dos centros de dados cloud para a periferia e entre componentes periféricos, por forma a otimizar métricas de desempenho e existir uma utilização eficiente dos recursos limitados dos componentes periféricos.

1.2 Problema

A heterogeneidade dos componentes periféricos, as interligações entre os micro-serviços que compõem as aplicações e a gestão eficiente das migrações/replicações dos micro-serviços origina um sistema bastante complexo ao nível da monitorização e gestão. O objetivo particular deste trabalho foca-se no aspeto da gestão desse sistema, considerando que as métricas necessárias para a avaliação dinâmica e decisão de migração/replicação de micro-serviços, são disponibilizadas por um subsistema de monitorização. Alguns exemplos de métricas são a latência dos pedidos aos micro-serviços, o nível de distribuição de carga pelos componentes periféricos e o custo associado à computação de informação e à comunicação entre dispositivos. Estes valores podem ser usados para fazer decisões quanto à migração/replicação de micro-serviços considerando, entre outros fatores, os recursos dos componentes periféricos, a sua distribuição geográfica e o volume de dados

na rede. Existem algumas características da computação edge e da arquitetura de micro-serviços que devem ser realçadas por serem fatores importantes na gestão deste sistema:

Hierarquia de nós. Os nós periféricos formam uma hierarquia entre os dispositivos *front-end* e os centros de dados cloud. A gestão do sistema e a decisão de migração/replicação de micro-serviços deve ter em conta, não só as métricas disponibilizadas pela monitorização, mas também a organização hierarquia dos nós na periferia.

Nós ao longo do *continuum* cloud/edge. A gestão do sistema deve considerar todos os nós na hierarquia. Só assim será possível atingir uma utilização eficiente dos recursos periféricos.

Dependência entre micro-serviços. Muitas vezes a decisão relativa a uma ação sobre um micro-serviço afeta um conjunto de outros micro-serviços que estão dependentes do primeiro. A gestão dos recursos na periferia deve ter em conta essas dependências, por forma a melhorar o desempenho do sistema. Com um conhecimento prévio mínimo, as dependências entre micro-serviços terão que ser detetadas dinamicamente.

1.3 Contribuições

A contribuição principal deste trabalho é a extensão de um sistema de gestão e coordenação automático para migração/replicação de micro-serviços entre componentes na periferia e centros de dados cloud. Este sistema foi iniciado pelo trabalho efetuado numa dissertação anterior [3]. A extensão do sistema contempla a:

Extensão da arquitetura. Incluindo mais métricas para além da latência, como por exemplo, a carga de tarefas e o custo da computação nos componentes da periferia. Também considerando uma hierarquia de nós que influencia a decisão de migração/replicação dos micro-serviços. E incluindo a dependência entre micro-serviços com o objetivo de melhorar o desempenho do sistema.

Extensão do middleware. As considerações adicionais na arquitetura permitem a extensão do middleware ao adicionar novas funcionalidades.

E para a avaliação das extensões realizadas no sistema:

Avaliaçao. O desenvolvimento e utilização de uma aplicação de demonstração (chamada *SockShop*) permite a verificação do desempenho e correção das extensões realizadas à arquitetura e ao middleware.

1.4 Organização do documento

Capítulo 2

Estado da Arte

O objetivo deste capitulo é apresentar os conceitos e definições necessários para compreender a proposta de trabalho disponibilizada no Capítulo 3.

2.1 Computação Cloud

2.1.1 Introdução à Cloud

A National Institute of Standards and Technology (NIST) define a computação cloud como sendo:

> um modelo para permitir acesso ubíquo, conveniente e a pedido a um conjunto de recursos de computação partilhados (Ex.: redes, servidores, armazenamento, aplicações e serviços) que possam ser rapidamente provisionados e libertados com um mínimo esforço de gestão ou interação do fornecedor do serviço [7].

A computação cloud surgiu nos primeiros anos do novo milénio. Foi um movimento motivado pelo facto do processamento de informação em grandes sistemas de computação e armazenamento ser mais económico e facilmente acessível através da Internet. Desta forma, o acesso à computação pode ser visto como um serviço público semelhante à distribuição de água ou eletricidade [**CCTP13a**]. Muitos fornecedores de serviços computacionais como a Google, Amazon, IBM e Microsoft estão atualmente a promover este paradigma como uma utilidade [9].

É um dos resultados do *Grid movement* que tinha como objetivo desenvolver uma computação em grelha (*Grid computing*) constituído num sistema distribuído com um grande número de sistemas heterogéneos e diferentes domínios administrativos [**b1**]. O facto de serem utilizados sistemas heterogéneos dificultou muito alguns, já difíceis, problemas de

gestão de sistemas como o agendamento, alocação de recursos, distribuição de carga e tolerância a falhas [**b1**]. Embora tenha sido um projeto ambicioso e muito popular entre as comunidades cientificas e engenharias, não abordou problemas relacionados com a industria e, portanto, não teve o impacto na industria da tecnologia da informação que se esperava [**b1**].

Aprendendo com as lições retiradas do *Grid movement*, uma estrutura de suporte à computação cloud é maioritariamente homogénea em termos de segurança, gestão de recursos e custos e tem como alvo a computação industrial [**b1**].

Antes do aumento de popularidade da computação cloud, os sistemas Peer-to-Peer (P2P) foram um dos grandes interesses da comunidade cientifica e industrial. Os dois modelos têm significantes diferenças, principalmente no facto dos sistemas P2P serem auto-organizados e descentralizados, enquanto que os servidores na cloud têm um único domínio administrativo e uma gestão central.

De acordo com a entidade NIST, a computação cloud tem cinco características essenciais, três diferentes modelos de serviço e quatro diferentes tipos [7].

As cinco características essenciais são, nomeadamente, *self-service* a pedido, amplo acesso à rede, agrupamento de recursos, rápida elasticidade e medição de serviço.

Self-service a pedido. Esta característica vem do facto do consumidor conseguir adquirir automaticamente, e após necessidade, recursos computacionais como o tempo de computação, o armazenamento ou capacidade de rede sem ser necessária a interação humana com os fornecedores de serviço.

Amplo acesso à rede. As capacidades da cloud estão disponíveis através da rede e são acedidas por plataformas heterogéneas de clientes através de mecanismos padrão.

Agrupamento de recursos. Os diferentes recursos físicos e virtuais dos fornecedores cloud são agrupados para dinamicamente fornecer múltiplos consumidores dependendo da procura. Existe uma independência da localização porque normalmente o consumidor não tem controlo nem tem conhecimento da localização exata dos recursos a serem utilizados, mas pode ser possível a identificação da localização mais geral como o continente, pais ou centro de dados.

Rápida elasticidade. Os recursos devem ser elasticamente provisionados e libertados, em alguns casos automaticamente, de modo a ser possível escalar rapidamente dependendo da procura do consumidor. Desta forma, para o cliente é dada uma visão de que os recursos disponíveis são ilimitados e que podem ser consumidos em qualquer quantidade a qualquer momento.

Medição de serviço. Os sistemas cloud controlam e otimizam automaticamente a utilização dos recursos através da monitorização de capacidades métricas do sistema (e.g. armazenamento, processamento, utilização de rede, número de consumidores ativos).

Na computação cloud é feita a distinção entre três modelos disponibilizados aos seus clientes, visualizados na figura 2.1. Estes modelos permitem isolar características para lidar com as diferentes necessidades empresariais, educacionais e sociais [9].

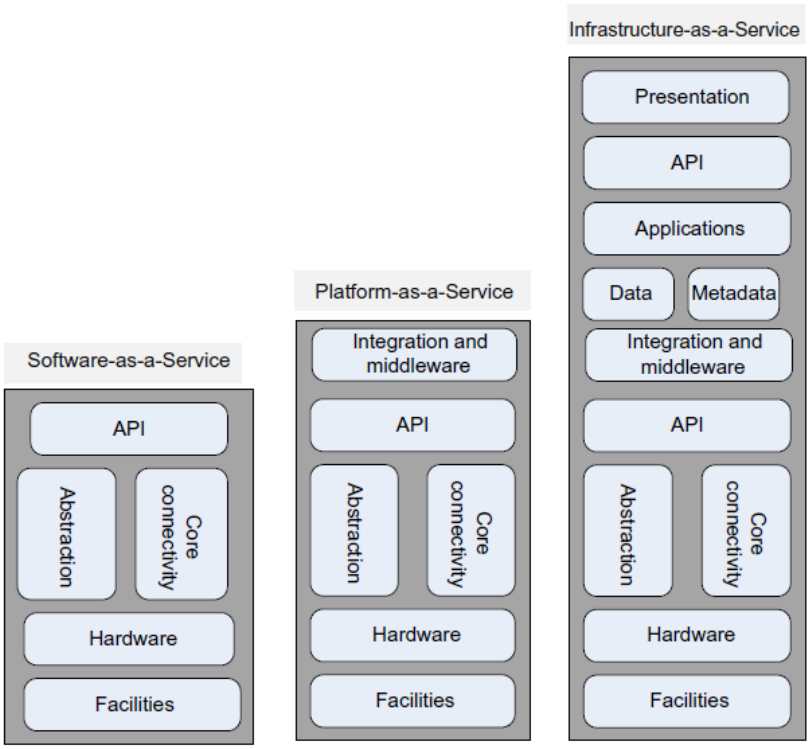

Figura 2.1: Os diferentes modelos presentes na computação cloud [**b1**].

O modelo Software as a Service (SaaS) fornece as capacidades necessárias para a utilização de aplicações disponibilizadas pelos fornecedores. O acesso a essas aplicações, por parte dos clientes, é feito através de uma interface bem definida. Neste modelo, o cliente não gere nem controla qualquer componente da infraestrutura cloud, apenas acede aos serviços através da interface a este disponibilizada. No modelo SaaS é comum o armazenamento dos dados ser efetuado noutro local relativamente longe da aplicação, não sendo, portanto, um modelo ideal para aplicações que não permitam o armazenamento externo dos dados. As características deste modelo, enumeradas anteriormente, também não são ideais para aplicações que requerem respostas em tempo real.

O modelo Platform as a Service (PaaS) é caracterizado por ter a capacidade de alojar aplicações criadas ou adquiridas pelos consumidores utilizando as ferramentas e linguagens de programação dos fornecedores cloud. Tal como no modelo SaaS, o cliente não gere os componentes da infraestrutura cloud, mas neste modelo existe controlo sobre as aplicações que desenvolve. Este modelo é particularmente bom na área de desenvolvimento de software para permitir colaboração entre vários utilizadores e, eventualmente, automatização do processo de desenvolvimento, lançamento e manutenção de aplicações. Aplicações que necessitem de ter um considerável nível de portabilidade, que utilizem linguagens de programação proprietárias, ou que necessitem do controlo da infraestrutura cloud, não são adequadas ao modelo PaaS.

O modelo Infrastructure as a Service (IaaS) fornece ao cliente um conjunto de recursos

presentes num sistema de computação, como, processamento, armazenamento e rede. Esses recursos a que o cliente tem acesso podem ser utilizados para executar o mais variado tipo de software. Tal como no modelo SaaS e PaaS, o cliente não gere os componentes da infraestrutura cloud, mas diferentemente dos outros dois modelos, o cliente tem controlo sobre o software que executa no sistema e que pode incluir sistemas operativos e/ou aplicações. Este modelo é utilizado principalmente para arrendamento de poder de computação utilizado para os mais diversos serviços.

Existem diferentes entidades envolvidas na computação cloud como mostra a figura 2.2.

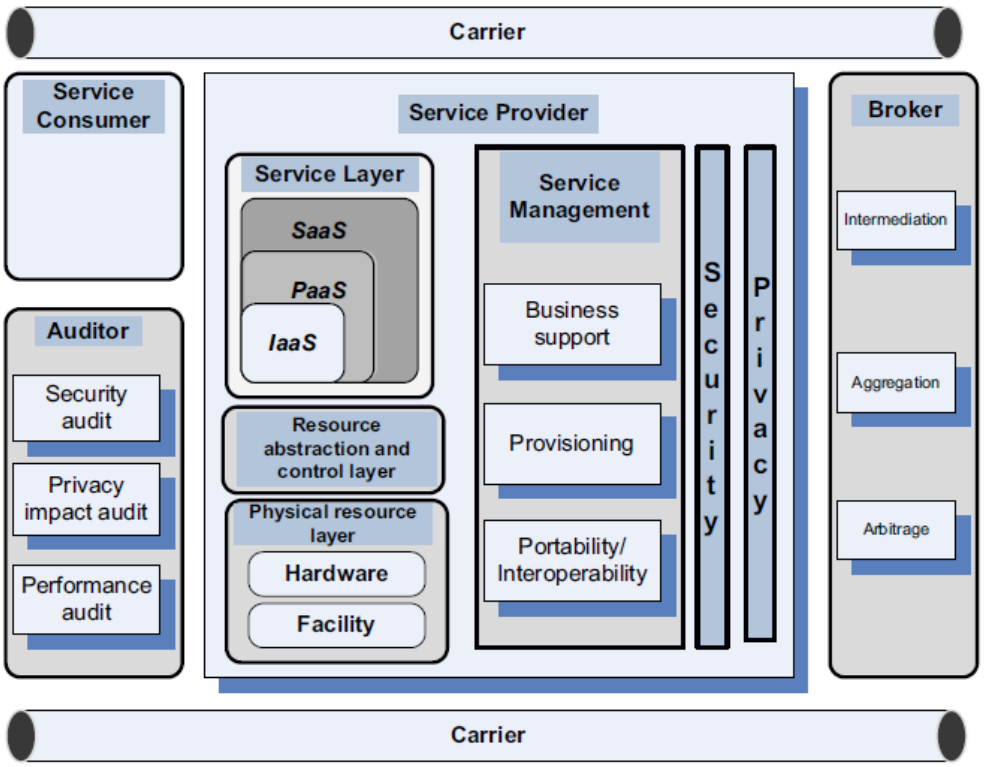

Figura 2.2: As diferentes entidades envolvidas na computação cloud [**b1**].

O consumidor do serviço (*Service Consumer*) utiliza os serviços dos fornecedores cloud mediante uma relação comercial. O fornecedor cloud (*Service Provider*) é a entidade responsável por disponibilizar o serviço cloud. O transportador (*Carrier*) atua como intermediário entre o fornecedor e consumidor, e é responsável por disponibilizar a conetividade e o transporte dos serviços cloud entre ambas as entidades. O corretor (*Broker*) é a entidade que gere a utilização, desempenho e garantias de entrega dos serviços cloud negociando as relações entre o consumidor e o fornecedor. O auditor (*Auditor*) tem a responsabilidade de avaliar os serviços cloud, o desempenho, a segurança e a implementação da cloud através de audições feitas ao sistema que avaliam e medem esse sistema de acordo com certos critérios.

A computação cloud pode ser divida em quatro tipos diferentes: as clouds privadas,

as clouds comunitárias, as clouds públicas e as clouds híbridas.

O tipo de computação cloud privada está adaptado e otimizado para ser utilizado por organizações, sendo que a sua infraestrutura é operada exclusivamente para uma organização.

A infraestrutura de uma cloud comunitária está adaptada para ser partilhada por várias organizações com requerimentos específicos.

As clouds públicas são utilizadas por organizações para venda de serviços cloud ao público em geral ou a um grupo industrial.

A infraestrutura das clouds híbridas é composta por dois ou mais dos tipos de computação cloud anteriormente referidos. Cada tipo envolvido é agregado através de tecnologia padrão ou proprietária para permitir portabilidade entre os diferentes tipos de cloud.

2.1.2 Vantagens da Computação Cloud

A computação cloud tem várias vantagens por ser um modelo realista, utilizar tecnologias avançadas e recentes, ser conveniente para o utilizador e ser económico.

O modelo da computação cloud é realista por ser homogéneo, tanto em hardware como em software, e ter um único domínio administrativo. Este foi um dos aspetos revistos e melhorados após o fracasso do, anteriormente referido, *Grid movement*. O facto de haver um sistema homogéneo e com um único domínio administrativo permite simplificar algumas soluções de problemas relacionados com sistemas de computação, como tolerância a falhas, qualidade de serviço, gestão de recursos ou segurança. Por exemplo, a criação de um conjunto de máquinas virtuais iguais e/ou similares, facilita o *deployment* das aplicações bem como as ações de gestão como atualizações, ou substituição em caso de falhas, nessas máquinas virtuais.

A computação cloud é desenvolvida e promovida por grandes empresas e grupos comerciais, com grandes capacidades económicas, o que permite vastos investimentos em software, hardware, armazenamento, processadores e redes. Tecnologia essa necessária para desenvolver novos e melhores sistemas de computação cloud e competir por uma melhor posição de mercado [6]. O facto dessas grandes empresas terem a capacidade para alcançar uma distribuição global dos seus centros de dados permite também uma maior proximidade aos clientes nas diferentes regiões. Como a cloud utiliza uma política de pagamento baseada na utilização dos seus recursos, evita que os utilizadores tenham que investir numa infraestrutura e efetuar a manutenção de uma sistema de larga escala [6, 10]. Nos grandes centros de computação, devido à economia de escala, os fornecedores cloud conseguem um melhor aproveitamento dos recursos na cloud obtendo um baixo custo marginal de administração e operação [6, 10]. Devido ao baixo custo de operação é possível que os utilizadores utilizem os serviços cloud a preço reduzido comparativamente a outros centros de menor dimensão. O modelo de negócio praticado na cloud e o resultado da economia de escala permite tornar a computação cloud cada vez mais popular. A popularidade é absolutamente crucial para que o negócio seja rentável visto

que existe um enorme investimento em infraestruturas. Com um negócio rentável, é do interesse dos fornecedores investirem cada vez mais em centros de dados.

A computação cloud é altamente escalável e elástica por ser consistida por um conjunto homogéneo de sistemas de computação com uma política de pagamento baseada na sua utilização. As aplicações com elevada computação e utilização de dados que se possam particionar podem beneficiar de escalabilidade horizontal para melhorar os seus tempos de execução. E uma aplicação com crescente número de utilizadores pode tirar proveito da elasticidade da cloud de modo a suportar maior carga adicional com esforço mínimo dos desenvolvedores da aplicação.

A computação cloud baseia-se num paradigma cliente-servidor e a maioria das aplicações cloud atualmente disponíveis tiram proveito de uma comunicação sem estado entre clientes e servidores. Numa comunição sem estado, cada pedido é independente entre si e não é guardada informação entre pedidos do mesmo cliente. A utilização deste tipo de servidores tem vários benefícios como a sua rapidez e facilidade no estabelecimento de comunicações, facilidade de recuperação de falhas, bem como uma maior simplicidade, escalabilidade e robustez comparada com servidores com estado.

2.1.3 Limitações e Desafios da Computação Cloud

Apesar das diversas vantagens e desenvolvimentos tecnológicos proporcionados pela computação cloud, ainda há obstáculos neste modelo que devem ser superados.

Um desses obstáculos é garantir disponibilidade de serviço por parte dos fornecedores de cloud. Como a cloud pode ser partilhada por múltiplos utilizadores, o facto de existirem muitos utilizadores a usar o serviço simultaneamente não pode afetar negativamente a disponibilidade de serviço. Uma das soluções, embora não perfeita do ponto de vista económico, é garantir que existam recursos suficientes para satisfazer a previsão do maior número de utilizadores simultaneamente a usar o sistema.

Outro problema associado com a computação cloud está relacionado com a dependência dos utilizadores ao fornecedores cloud. Um utilizador ao usar um dos fornecedores cloud fica dependente do mesmo, sendo difícil a sua mudança para outro fornecedor. Uma solução atualmente em curso, mas não completa, passa por padronizar as tecnologias de modo a permitir uma menor dependência ao fornecedor cloud.

Certas aplicações na cloud utilizam muitos dados ficando bastante dependentes da velocidade de transferência de dados, o que pode causar um ponto de *bottleneck* no sistema. Uma das estratégias implementadas para aliviar o problema é armazenar a informação o mais perto possível do local onde é necessária. Quando a velocidade de transferência na rede é relativamente baixa, uma opção mais rápida e barata passa por armazenar os dados em memória não volátil e enviar os dados através de um método offline. À medida que a velocidade média de transferência dos dados aumenta ao longo dos anos, esta necessidade vai deixando de ser um problema.

Como dito anteriormente, a cloud é baseada na partilha de recursos para permitir uma

política de pagamento utilitária. Por vezes, esta partilha de recursos pode ser problemática no aspeto da previsibilidade do desempenho esperado. Para permitir a elasticidade e escalabilidade da computação cloud, são necessários novos algoritmos para o controlo de alocação de recursos e distribuição de carga capazes de escalar rapidamente. Uma das fortes apostas na resolução deste problema passa por utilizar Computação Automática baseada em organização e gestão própria [].

Outros dois problemas que perduram e ainda sem soluções universalmente aceites são o licenciamento de software e erros no sistema cloud. A atual tecnologia de gestão de licenciamento de software foi desenvolvida para ser utilizada em serviços não distribuidos. A computação cloud fornece um serviço distribuido, não sendo a tecnologia de licenciamento de software atual adaptável às necessidades da infraestrutura que constituem a cloud.

A infraestrutura cloud é complexa envolvendo múltiplos componentes o que torna a deteção de erros no sistema extremamente complexa e difícil. Outra razão para a dificuldade na deteção de erros deve-se ao facto do sistema poder envolver várias organizações com barreiras de segurança pouco definidas, um processo chamado *de-perimeterisation*. É também um problema para a determinação da responsabilidade de erros devido à cadeia complexa de eventos, em diferentes entidades, que muitas vezes é necessária para desencadear o erro. Com a responsabilidade do erro a ser partilhada por várias entidades, muitas vezes é difícil responsabilizar o conjunto de entidades pelo erro causado.

A computação cloud vem reforçar ainda mais alguns receios relacionados com a ética computacional. Como o controlo da computação passa a ser delegado a um serviço de terceiros, existe maior risco potencial para situações de acesso não autorizado.

A computação cloud aumenta o armazenamento e a circulação de dados pessoais entre entidades o que agrava problemas relacionados com roubo de identidade devido ao acesso indevido dessa informação pessoal. Este facto pode colocar em causa o sucesso da computação cloud devido ao aumento de desconfiança da sociedade perante a (in)segurança deste modelo. É do conhecimento da sociedade que os fornecedores cloud tenham armazenado uma enorme quantidade de sensíveis dados pessoais. A confiabilidade e auditabilidade dos dados é um importante problema a superar. Os recentes acontecimentos e acusações relacionados com a venda de dados pessoais por parte de empresas tecnológicas gigantes, caso do Facebook [**b2**], impõe um grande obstáculo à aceitação da computação cloud como um método viável e seguro para o armazenamento de dados pessoais. E, para complicar mais a situação, a privacidade é afetada por diferenças culturais. Há culturas que favorecem a partilha enquanto outras favorecem a privacidade. A computação cloud pretende ser global e portanto devem ser discutidas soluções para este tipo de problemas culturais.

O desafio principal da computação cloud é mesmo a segurança. Talvez seja irrealista esperar um modelo de segurança igual em todos os tipos e modelos cloud. Uma cloud pública dificilmente consegue ter um ambiente adequado para todo o tipo de aplicações. Existem aplicações mais sensíveis, como na área bancária, militar ou de saúde, que

necessitam de uma maior segurança. A utilização de clouds híbridas pode ajudar a solucionar este problema, usando clouds públicas para algum tipo de processamento e clouds privadas para lidar com a informação mais sensível.

É critico para o sucesso da computação cloud que seja ganha a confiança da sociedade porque a cloud necessita de um grande número de utilização para que seja viável o grande investimento em infraestruturas feito pelas companhias fornecedoras de cloud.

Atualmente, o financiamento da pesquisa da computação cloud está mais direcionado para a geração de lucro do que para a regulamentação. É óbvia a necessidade da intervenção de entidades reguladoras para a regulação da computação cloud por forma a existir uma maior ética na relação fornecedor-consumidor e ser obtida uma maior aceitação por parte da sociedade a este novo modelo de computação. Um dos principais fatores na regulação da computação cloud está relacionado com o registo das atividades e das ações, permissões e responsabilidades das entidades envolvidas. Os registos permitem obter uma evidencia clara que pode ser utilizada em eventuais problemas.

Apenas grandes companhias conseguem ter o poder económico para investir em infraestruturas cloud. Um outro problema está relacionado com o facto de apenas existirem algumas companhias que dominam o mercado. Existem preocupações sérias relacionadas com a manipulação de preços e políticas.

A elasticidade da computação cloud requer a habilidade de distribuir as computações e os dados por vários sistemas. A coordenação entre esses sistemas é um dos grandes obstáculos num sistema distribuído e, mais especificamente, na computação cloud.

Com ainda importantes e difíceis problemas para serem superados, o futuro sucesso da computação cloud está dependente da capacidade de promoção da computação utilitária por parte das companhias e centros de investigação para convencer uma maior população das vantagens da computação centrada em rede e conteúdo. É necessário encontrar soluções para aspetos críticos de disponibilidade e qualidade de serviço, escalabilidade e elasticidade, segurança e confiabilidade.

2.1.4 Casos de estudo: Amazon e Google

2.2 Computação Edge

Na última década tem existido uma centralização e consolidação de serviços e aplicações em centros de dados originando o conceito de computação cloud [8]. A computação cloud mudou radicalmente quase todos os aspetos da vida humana [5]. Os benefícios económicos da computação cloud referidos na Subseção 2.1.2 e o atual enorme investimento em recursos e desenvolvimento da cloud sugerem que a computação cloud irá ser uma característica presente na computação futura [10]. Mas, as aplicações baseadas na cloud utilizam servidores centralizados num centro de dados para processar dados, que são gerados por dispositivos na periferia. Devido aos recentes avanços tecnológicos e a novos tipos de aplicações, é insuportável continuar a enviar e receber toda a informação produzida

e necessária para um centro de dados central. Com o aumento rápido do número desses dispositivos é colocada cada vez maior pressão nas estruturas computacionais cloud e na comunicação com os centros de dados. Isso pode ter efeitos adversos na qualidade do serviço e na experiência do utilizador [1]. Também a preocupação crescente com problemas relacionados com a confiança, privacidade e autonomia associados à computação cloud sugere claramente a necessidade de uma mudança de paradigma.

A computação edge é um novo paradigma promissor de computação cloud descentralizado que coloca a aquisição de dados e funções de controlo, o armazenamento de conteúdo com grande utilização de rede, e as aplicações, mais perto do utilizador final. São utilizados dispositivos na periferia da rede (Internet ou rede privada) ligados a uma arquitetura de computação cloud com maior capacidade de recursos. [2]. É possível assim mover alguma carga computacional e aplicações dos centros de dados centralizados para a periferia da rede, de modo a aproveitar melhor os recursos computacionais atualmente pouco utilizados [1]. Deste modo, a computação edge atua como camada intermédia entre o utilizador e os centros de dados cloud para oferecer um serviço com menor latência de rede [5]. Os dispositivos na periferia complementam as computações realizadas na cloud [1].

A origem da computação edge pode ser associada ao aparecimento de Content Delivery Network (CDN) que utiliza nós na periferia para melhor o desempenho da web ao fazer *cache* e *pre-fetch* de, principalmente, conteúdo com grande utilização de rede como é o caso do conteúdo multimédia. A computação edge generaliza esse conceito ao ser integrado numa infraestrutura de computação cloud distribuída. E em vez de apenas fazer *cache* de conteúdo web, o objetivo é ser possível executar código arbitrário nos nós periféricos [10]. Uma das possibilidades para permitir a execução de código arbitrário em dispositivos que normalmente estão adaptados a uma certa carga de trabalho é a utilização de virtualização, discutida na Seção 2.4. O código a executar pode ser encapsulado em máquinas virtuais ou *containers* para existir isolação, segurança e gestão de recursos [10].

É possível distinguir na figura 2.3 três tipos de aplicações na periferia: aplicações on-premises, aplicações para agregação e controlo de dados IoT e distribuição de conteúdo com elevada utilização de rede.

A garantia de disponibilidade é uma grande preocupação das aplicações on-premises (Ex.: controlo industrial ou institucional). A utilização de componentes na periferia pode ajudar a superar desafios de disponibilidade no caso de, por exemplo, falha de energia prolongada ou um ataque Distributed Denial of Service (DDoS). As companhias que estão atualmente a utilizar a cloud no seu negócio, podem beneficiar da computação edge para aumentar a disponibilidade do seu sistema ao introduzir redundância das suas aplicações criticas em componentes periféricos. E aumentar a segurança dos dados, visto que podem ser processados e/ou filtrados próximo onde são recolhidos [2].

As tecnologias associadas ao fenómeno de tornar tudo inteligente (Ex.: cidades, agricultura, automóveis e saúde) requerem a utilização de grandes quantidades de sensores IoT [2]. As soluções atuais requerem a migração dos dados para centros de dados cloud

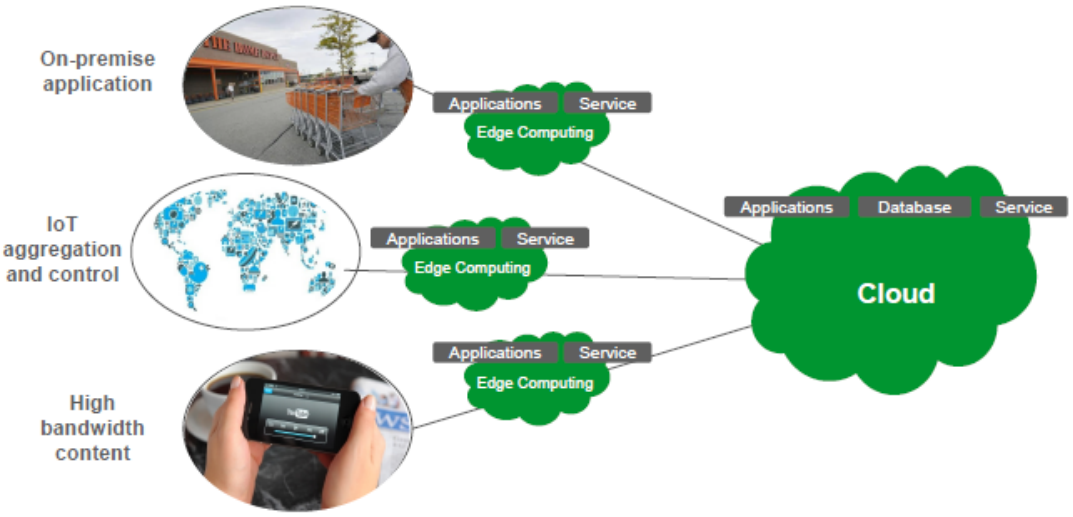

Figura 2.3: Representação da interação dos dispositivos na periferia com a computação edge e cloud [2].

para serem interpretados. O que implica uma latência grande, incomportável para aplicações de tempo real. E a grande quantidade de dados gerados por estes sensores IoT não é compatível com o paradigma da computação cloud porque a quantidade é de tal maneira elevada, que se torna incomportável transportar todos esses dados pela rede. As aplicações para agregação e controlo de dados IoT podem beneficiar da estrutura que compõem a computação edge para reduzir a latência devido à maior proximidade aos componentes na periferia. E ao processar e/ou filtrar os dados gerados pelos sensores IoT nesses componentes periféricos, é possível reduzir a quantidade de dados transmitidos para os centros de dados cloud.

As aplicações que requerem utilização elevada de rede (Ex: VOD, 4k Tv, video streaming) são as mais prováveis para congestionar a rede. Por forma a aliviar essa congestão, por parte dessas aplicações, os fornecedores do serviço estão a interligar um sistema de computadores com capacidade para disponibilizar mais rapidamente o conteúdo ao utilizador com menor congestão da rede. Esse sistema de computadores, visualizado na figura 2.4, que fazem *cache* do conteúdo, são um exemplo de computação edge [2].

Avanços tecnológicos nos dispositivos perto dos utilizadores (Ex.: dispositivos inteligentes, móveis e *wearables*, sensores, nano-centro de dados) e a previsão do aumento da sua utilização podem ajudar a implementar essa mudança de paradigma devolvendo o controlo das aplicações, dados e serviços à periferia da Internet [8]. Esses avanços tecnológicos estão a permitir visionar novos tipos de aplicações aplicáveis em, por exemplo, cidades inteligentes, cuidados de saúde pervasivos, realidade aumentada, multimédia interativa, IoT e assistência cognitiva [5]. Mas, essas novas aplicações têm todas um problema em comum, são extremamente sensíveis à latência [5]. Outro problema está relacionado com a enorme quantidade de dados produzidos e usados por esse tipo de aplicações.

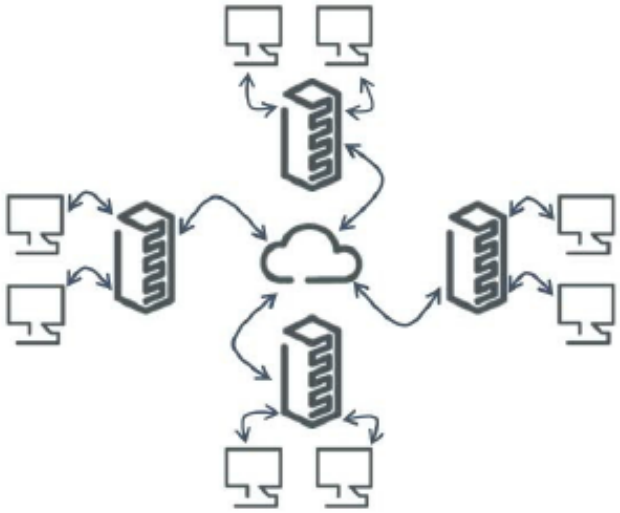

Figura 2.4: Sistema de computadores para disponibilizar o conteúdo mais perto do utilizador [2].

Um dos aspetos interessantes deste novo paradigma está relacionado com o facto da fronteira entre homem e máquina ficar menos definida. Ou seja, os humanos fazem parte da computação e das decisões relacionadas o que origina sistemas desenhados com foco nos humanos (*human-centered system design)*. Esta visão onde o aspeto principal é o papel fundamental dos humanos é referida por [8] como sendo a *Edge-centric Computing*.

A computação P2P, conceito introduzido por volta dos anos 2000 com o aparecimento de sistemas de partilha de ficheiros, está bastante relacionado com a computação edge. O paradigma *Edge-centric Computing* origina do P2P, mas em vez de tentar uma descentralização completa do sistema, expande o conceito de *peer* para todos os dispositivos na periferia da Internet e combina a computação P2P com a cloud.

Dois exemplos de *Edge-centric Computing* são a Mobile Edge Computing (MEC) e Mobile Cloud Computing (MCC) [9]. A MEC combina a operação de servidores periféricos com estações base para melhorar a eficiência da rede e a utilização de serviços considerando dispositivos móveis. A MCC pretende auxiliar na execução de aplicações nos dispositivos móveis usando nano-centros de dados (*cloudlets*) para deslocar a carga de tarefas dos dispositivos móveis para outros dispositivos com menos restrições de recursos (energia, armazenamento, computação). Os *cloudlets* fazem a intermediação entre os dispositivos móveis e os servidores na cloud permitindo melhorar a Quality of Experience (QoE) dos utilizadores [9].

Outro tipo de computação na periferia é a computação fog. Enquanto que a computação edge foca-se apenas na rede periférica, a computação fog procura usar tanto a rede na periferia como o núcleo da rede (Ex.: routers principais, servidores regionais, wan switches). Este tipo de computação é particularmente bom para ser usado por dispositivos IoT porque os componentes fog podem ser colocados ao alcance desses dispositivos

resultando numa significativa redução da latência. Ao contrário da computação edge, a computação fog tem as capacidades para estender os serviços base da cloud (IaaS, PaaS, SaaS) [9].

Na Figura 2.5 estão ilustrados todos os tópicos relevantes associados à computação edge/fog.

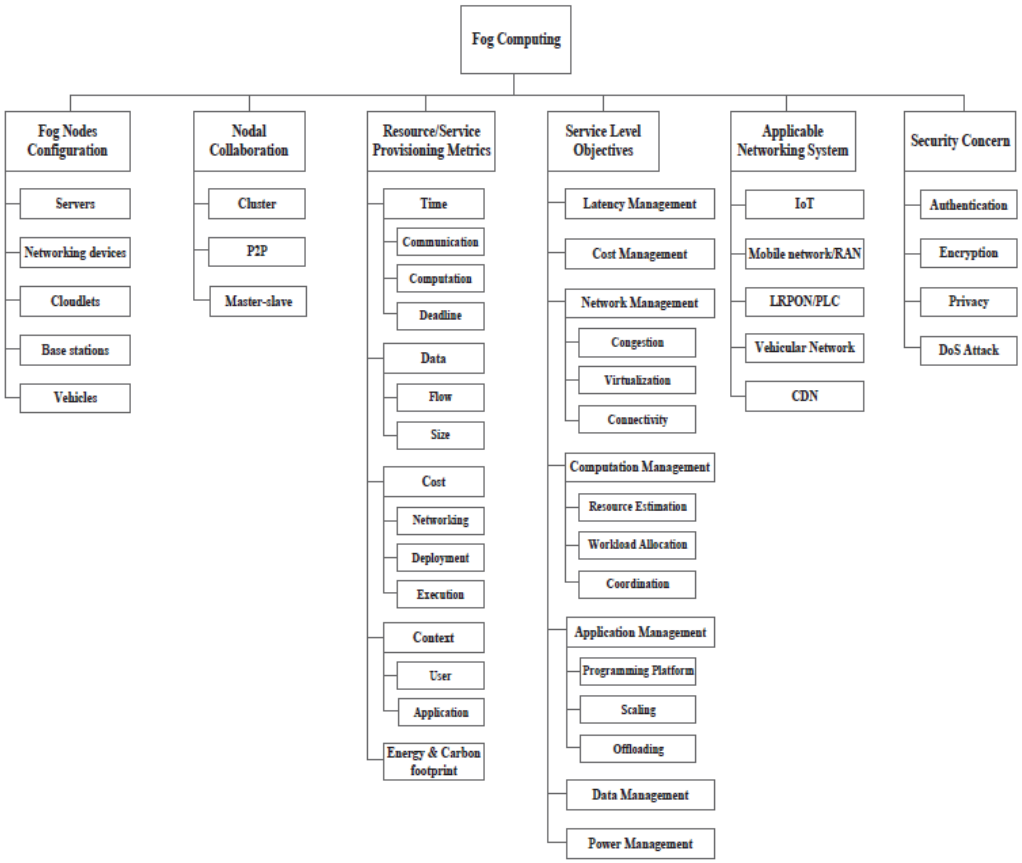

Figura 2.5: Taxonomia da computação edge/fog [9].

Na figura 2.6 estão visualizados os vários tipos de domínios de computação edge anteriormente abordados.

Podem ser considerados vários dispositivos na periferia da Internet que podem ser usados para suportar a computação edge como, por exemplo, *routers*, *gateways*, *switchs*, e estações base [5]. Ou componentes mais especializados como os dispositivos locais e centros de dados localizados e regionais [2]. Na figura 2.7 é possível a visualização da hierarquia que compõe a computação edge/fog. Os dispositivos edge/fog permitem estabelecer um ponto intermédio entre os dispositivos dos utilizadores e os centros de dados cloud.

2.2.1 Motivações e Vantagens da Computação Edge

TODO: melhorar o texto por forma a explicar a ordem pela qual aparece a informação

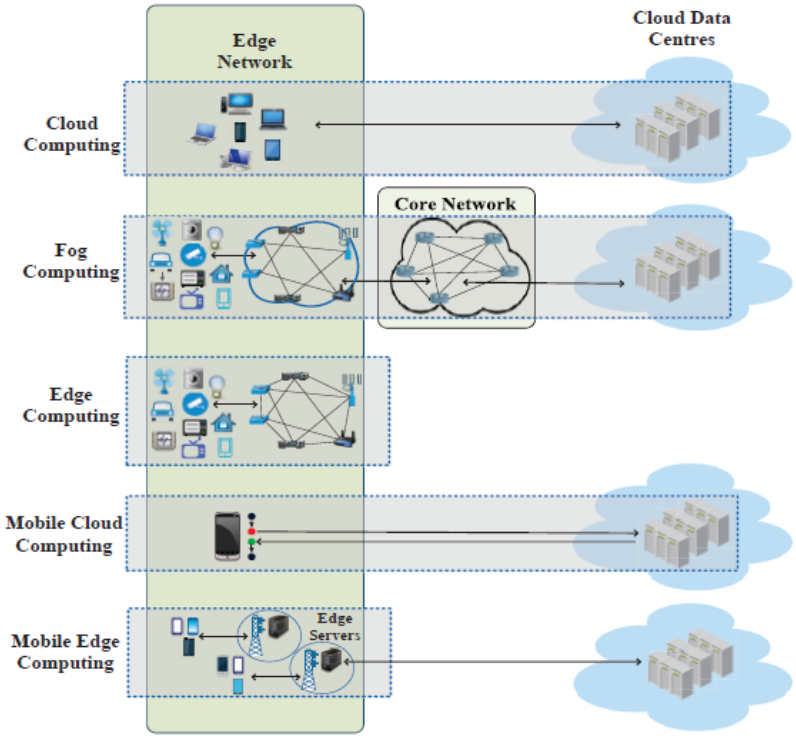

Figura 2.6: Os vários domínios da computação: cloud, fog, edge, mobile cloud e mobile edge [9].

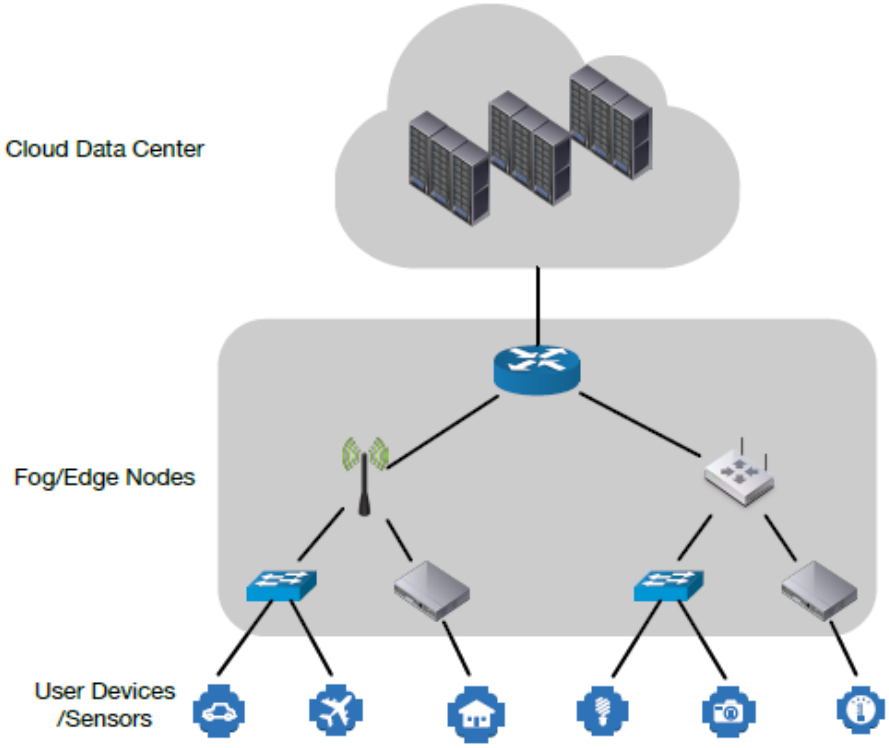

Figura 2.7: O modelo de computação edge/fog apresentando os recursos na periferia. Os microsserviços poderão ser migrados para qualquer dispositivo edge/fog [4].

Podem ser identificados alguns problemas importantes na computação cloud que a computação edge pretende solucionar, ou pelo menos, diminuir o seu impacto [8].

Recuperação do controlo das aplicações alojadas na cloud. A computação edge pode devolver o controlo das aplicações e dos sistemas aos dispositivos periféricos. Com um desenvolvimento suficiente da segurança na computação edge, poderá deixar de ser necessário que os utilizadores confiem unilateralmente numa entidade, como acontece no caso da computação cloud com o fornecedor cloud.

Maior privacidade de dados pessoais e privados. Atualmente, os dados pessoais são partilhados a serviços centralizados como sites e-commerce, serviços de classificação, motores de pesquisa, redes sociais e serviços de localização. A computação edge introduz a oportunidade de filtrar, nos dispositivos periféricos, os dados pessoais e privados. Isto permite uma maior segurança e maior controlo dos dados enviados para os centros de dados cloud.

Oportunidade de utilização de recursos. Com a utilização de um paradigma centralizado, está se a perder a oportunidade de exploração do grande potencial de computação, comunicação e poder de armazenamento presente nos dispositivos da periferia da rede. A computação edge pode fazer um melhor aproveitamento desses recursos ao migrar parte da computação para a periferia.

As principais motivações para uma mudança de paradigma centralizado (computação cloud) para um paradigma periférico (computação edge) são a diminuição do tempo de transporte de dados (latência), a diminuição de transferência de dados e o aumento da disponibilidade [2, 5]. Esta mudança é particularmente importante sabendo que a utilização da Internet está cada vez mais associada a conteúdo com elevada utilização de rede. Esse tipo de aplicações (Ex: VOD, 4k Tv, video streaming) contribuem para a congestão na rede porque recolhem um elevado volume de dados que é tipicamente processado na cloud. A computação edge coloca esse conteúdo e aplicações sensíveis à latência em dispositivos com poder computacional e com capacidades de armazenamento mais perto dos utilizadores [2].

A principal razão para o aparecimento da computação edge é aproximar o nível de poder computacional atualmente presente em centros de dados centralizados aos utilizadores. Ao explorar novas aplicações na área da computação móvel e IoT, é óbvia a importância dessa aproximação porque afeta a latência e limita a largura de banda. Mesmo utilizando uma conexão direta com tecnologia fibra, a latência está limitada devido à velocidade da luz. E utilizar uma estratégia *multihop* para cobrir uma grande área usando muitos pontos de acesso limita também a latência e a largura de banda porque cada salto na rede aumenta o tempo de encaminhamento [10]. Este fenómeno pode ser observado na Figura 2.8.

A proximidade dos nós periféricos, referidos por [5, 9, 10] como cloudlets, permite a existência de serviços cloud mais responsivo. Nova tecnologia como realidade virtual e aumentada pode beneficiar da computação edge ao processar a informação em cloudlets próximos sem afetar a interatividade e a experiência do utilizador. A proximidade dos

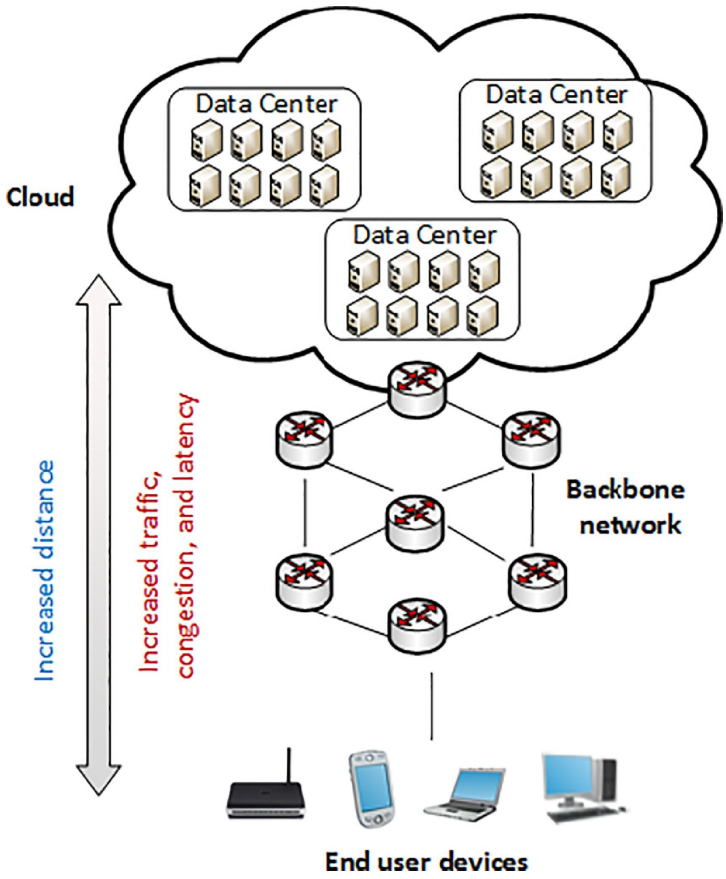

Figura 2.8: Problema da distância na computação cloud [5].

cloudlets permite existir baixa latência, maior largura de banda e baixa instabilidade entre o utilizador e o dispositivo que faz a computação [10].

Os dispositivos usados por utilizadores (Ex.: computadores pessoais, telemóveis, tablets, dispositivos *wearables*) têm hardware relativamente limitado comparado com os servidores nos centros de dados. Os dados gerados por esses dispositivos normalmente têm que ser enviados para a cloud para serem processados. Mas, às vezes nem toda a informação enviada é necessária para construir o resultado enviado ao cliente. Existem dados que podem ser filtrados ou analisados em nós na periferia. Apenas a informação e os metadados extraídos após a analise precisam de ser enviados para a cloud aliviando a comunicação entre os dispositivos *front-end* e os centros de dados [1, 10].

A rápida expansão da utilização de serviços cloud tem como consequência inevitável o consumo crescente de energia por parte dos centros de dados cloud. Existe uma grande necessidade de adotar estratégias eficientes de consumo energético, não só devido a preocupações monetárias, mas também ambientais. A computação edge permite incorporar uma gestão sensível de consumo de energia ao executar parte, ou toda, da computação necessária às aplicações e utilizadores, mais perto onde os resultados são usados [1]. O facto de ser reduzido o tráfego na rede a longa distância entre os utilizadores e os centros de dados cloud permite a existência de uma maior eficácia energética [5].

Funcionando como primeiro ponto de contacto na infraestrutura de cloud distribuída, um cloudlet pode restringir os dados que são enviados para a cloud como forma de garantir a sua privacidade [10].

Se um serviço cloud ficar indisponível devido a falhas de rede, falhas técnicas na cloud ou ataques DDoS, a utilização de cloudlets permite a ação de serviços que escondam temporariamente essa falha [10].

2.2.2 Desafios e Limitações da Computação Edge

A computação edge ainda é muito recente e as *frameworks* atualmente públicas, que facilitem o processo de desenvolvimento, ainda são muito limitadas. Esse tipo de *framework* deve satisfazer requerimentos como, por exemplo, o desenvolvimento de aplicações que processem pedidos em tempo real nos nós periféricos. A implementação de processamento de dados em tempo real na periferia da rede ainda é um campo de estudo em aberto. Outro requerimento é a facilitação do *deployment* de aplicações em nós periféricos. É preciso saber onde colocar a carga da aplicação, estudar políticas de conexão e saber que nós utilizar de modo a otimizar a utilização da computação edge. Para desenvolver tal *framework*, é necessário considerar alguns desafios a nível do *hardware*, *middleware* e *software* [1].

Heterogeneidade de componentes. Muitos nós computacionais localizados na periferia da rede (Ex.: pontos de acesso, estações base, *gateways*, pontos de agregação de tráfico) têm características diferentes e funções especificas para a carga de trabalho que suportam. O objetivo da computação edge é suportar qualquer tipo de computação e muitos nós na periferia estão adaptados para suportar apenas um certo tipo de computações [1, 9]. Um dos desafios passa por estudar como utilizar esses recursos na periferia de modo a suportar computação mais genérica. Usar técnicas de virtualização, como é o exemplo dos *containers* (Ex.: docker e kubernetes, abordados na Seção 2.4) são sérios candidatos para superar este desafio.

Descoberta de nós periféricos. De modo a explorar a periferia da rede são necessários novos mecanismos de descoberta para encontrar os nós que possam ser utilizados numa arquitetura de cloud descentralizada. Os métodos terão que ser bastante rápidos na identificação da disponibilidade e capacidade de recursos na periferia sem aumentar a latência ou comprometer a experiência do utilizador [1, 9].

Particionamento de tarefas e carga. Um outro desafio vem do facto das tarefas e da carga terem que ser particionadas pelos nós na periferia. O particionamento eficiente e automático das tarefas, sem necessidade de indicar explicitamente as capacidades individuais de cada nó, é um grande desafio da computação edge. É necessário o desenvolvimento de novas ferramentas de agendamento que particionem as tarefas pelos nós disponíveis [1, 9].

Garantia de qualidade de serviço. O quatro desafio está relacionado com a garantia da qualidade de serviço dos nós na periferia. Esses nós terão que assegurar uma execução

confiável das cargas de trabalho inicialmente pretendidas. A carga adicional de trabalho proveniente de dispositivos *front-end*, de outros nós periféricos ou da cloud, não pode comprometer de forma alguma o desempenho desses nós na execução das suas próprias tarefas [1, 9]. Na computação fog, o Service Level Agreement (SLA)[1] é afetado por diversos fatores (Ex.: custo de serviço, utilização de energia, características da aplicação, fluxo de dados, estado de rede). Portanto, num ambiente fog é bastante difícil especificar as medidas do serviço e os correspondentes Service Level Objectives (SLOs)[2] [9].

Segurança. A utilização pública e segura de nós na periferia é outro desafio da computação edge. É necessária uma definição clara dos riscos associados à utilização desses nós periféricos, tanto para os seus donos como para os seus utilizadores. A tecnologia relacionada com *containers* é um potencial candidato para uma utilização de nós na periferia com sucesso. Mas, ainda tem características de segurança pouco robustas que devem ser mais desenvolvidas [1, 9]. Devido à enorme quantidade de tipos de dados, escala física, frequência de comunicação e variedade de utilizadores, é difícil definir as considerações relativas à segurança na computação edge. Uma variável importante a ter em conta será a definição de segurança imposta pelos donos dos dispositivos periféricos que pode variar consoante as suas próprias necessidades e disposições [11].

Desenvolvimento de *frameworks*. Como os nós na periferia não têm todos os mesmos recursos, é necessário desenhar uma plataforma para desenvolvimento de aplicações distribuídas que tenha políticas de distribuição de tarefas pelos dispositivos periféricos e ajude na visualização de dados.

Novas tecnologias e definições padrão. O nível de padronização nas várias camadas (Ex.: infraestrutura, identificação, comunicação, descoberta, gestão, semântica e segurança) da computação edge e dispositivos IoT ainda está pouco definido [11]. É necessário haver um esforço conjunto das entidades reguladoras para desenvolver novos padrões robustos devido à heterogeneidade dos componentes periféricos. Um dos aspetos importantes a definir é a forma de comunicação entre os dispositivos *front-end* usados pelos utilizadores e os dispositivos IoT com os nós periféricos que fazem a ligação aos centros de dados centralizados. As tecnologias de comunicação têm tido bastante evolução recentemente procurando melhorar a velocidade e confiança dos vários métodos de transmissão. Um dos padrões que se destaca são as especificações da família 802.x definidos pelo Institute of Electrical and Electronics Engineers (IEEE) [11]. Os desenvolvimentos recentes pretendem abordar problemas relacionados com a comunicação na periferia da Internet. Na área de troca de mensagens, pode ser usado o habitual Hyper Text Transfer Protocol (HTTP)/Hyper Text Transfer Protocol Secure (HTTPS). Embora devem ser considerados outros métodos mais especializados, que têm as suas vantagens, como o Message Queuing Telemetry Transport (MQTT), o Constrained Application Protocol (CoAP), o Advanced Message Queuing Protocol (AMQP) e o Data Distribution Service (DDS) [11].

[1] Especificação clara dos serviços que o cliente pode esperar do fornecedor.
[2] Características mensuráveis específicas associadas a um SLA.

2.3 Microsserviços

2.3.1 Monitorização

2.4 Virtualização

Capítulo 3

Proposta de Solução

Bibliografia

[1] S. B.P.K.D.S. N. Blesson Varghese Nan Wang. "Challenges and Opportunities in Edge Computing". Em: *2016 IEEE International Conference on Smart Cloud (SmartCloud)*. New York, NY, USA: IEEE, nov. de 2016. ISBN: 978-1-5090-5263-9. DOI: https://doi.org/10.1109/SmartCloud.2016.18. URL: https://ieeexplore.ieee.org/abstract/document/7796149/.

[2] S. Carlini. *The Drivers and Benefits of Edge Computing*. Rel. téc. 226. Schneider Electric's Data Center Science Center, fev. de 2019. URL: https://www.schneider-electric.com/en/download/document/APC_VAVR-A4M867_EN/.

[3] A. V. Carrusca. "Gestão de micro-serviços na Cloud e Edge". Tese de mestrado. Calçada de Alfazina 2, 2825-149 Caparica: Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, set. de 2018.

[4] B. V. Cheol-Ho Hong. *Resource Management in Fog/Edge Computing: A Survey*. Cornell University on Distributed, Parallel, and Cluster Computing (cs.DC), Ithaca, NY 14850, EUA. Set. de 2018. URL: https://arxiv.org/abs/1810.00305.

[5] A. E.S.U. K. Kashif Bilal Osman Khalid. "Potentials, trends, and prospects in edge technologies: Fog, cloudlet, mobile edge, and micro data centers". Em: *Computer Networks*. Vol. 130. Elsevier, jan. de 2018, pp. 94–120. DOI: https://doi.org/10.1016/j.comnet.2017.10.002. URL: https://www.sciencedirect.com/science/article/pii/S1389128617303778.

[6] D. C. Marinescu. "Cloud Computing: Theory and Practice". Em: First. Morgan Kaufmann, 2013, pp. 0–0. ISBN: 978-0-12404-627-6.

[7] P. Mell e T. Grance. *The NIST Definition of Cloud Computing*. Rel. téc. 800-145. Computer Security Division, Information Technology Laboratory, National Institute of Standards e Technology, Gaithersburg, MD 20899-8930: National Institute of Standards e Technology, set. de 2011.

[8] D. E.A.D.T.H.A.I.M.B.P.F.E. R. Pedro Garcia Lopez Alberto Montresor. "Edge-centric Computing: Vision and Challenges". Em: *ACM SIGCOMM Computer Communication Review*. Vol. 45. 5. ACM, out. de 2015, pp. 37–42. DOI: https://doi.org/10.1145/2831347.2831354. URL: https://dl.acm.org/citation.cfm?id=2831354.

[9] R. B. Redowan Mahmud Ramamohanarao Kotagiri. "Fog Computing: A Taxonomy, Survey and Future Directions". Em: *Internet of Everything. Internet of Things (Technology, Communications and Computing)*. Out. de 2017, pp. 103–130. DOI: https://doi.org/10.1007/978-981-10-5861-5_5. URL: https://link.springer.com/chapter/10.1007%2F978-981-10-5861-5_5.

[10] M. Satyanarayanan. "The Emergence of Edge Computing". Em: *Computer*. Vol. 50. IEEE, jan. de 2017, pp. 30–39. DOI: https://doi.org/10.1109/MC.2017.9. URL: https://ieeexplore.ieee.org/document/7807196.

[11] A. Sill. "Standards at the Edge of the Cloud". Em: *IEEE Cloud Computing*. IEEE, abr. de 2017, pp. 63–67. DOI: https://doi.org/10.1109/MCC.2017.23. URL: https://ieeexplore.ieee.org/document/7912167.