

Roicenir Girardi Rostirolla

# **Computação em nuvem: um estudo orientado para aquisição em órgãos públicos**

São José - SC

fevereiro/2023

Roicenir Girardi Rostirolla

## **Computação em nuvem: um estudo orientado para aquisição em órgãos públicos**

Monografia apresentada à Coordenação de Engenharia de Telecomunicações do Instituto Federal de Santa Catarina para a obtenção do diploma Bacharel em Engenharia de Telecomunicações.

Instituto Federal de Santa Catarina – IFSC

Campus São José

Engenharia de Telecomunicações

Orientador: Prof. Ederson Torresini Me.

Coorientador: Profa. Elen Macedo Lobato Dra.

São José - SC

fevereiro/2023

Roicenir Girardi Rostirolla

Computação em nuvem: um estudo orientado para aquisição em órgãos públicos/  
Roicenir Girardi Rostirolla. – São José - SC, fevereiro/2023-

68 p.

Orientador: Prof. Ederson Torresini Me.

Coorientador: Profa. Elen Macedo Lobato Dra.

Monografia (Graduação) – Instituto Federal de Santa Catarina – IFSC  
Campus São José

Engenharia de Telecomunicações, fevereiro/2023.

1. Computação em nuvem. 2. Contratação pública. 3. Cloud computing. I. Ederson Torresini. II. Instituto Federal de Santa Catarina. III. Campus São José. IV. Computação em nuvem: um estudo orientado para aquisição em Órgãos Públicos

Roicenir Girardi Rostirolla

## **Computação em nuvem: um estudo orientado para aquisição em órgãos públicos**

Monografia apresentada à Coordenação de Engenharia de Telecomunicações do Instituto Federal de Santa Catarina para a obtenção do diploma Bacharel em Engenharia de Telecomunicações.

Trabalho aprovado. São José - SC, 10 de fevereiro de 2023:

---

**Prof. Ederson Torresini Me.**  
Orientador  
Instituto Federal de Santa Catarina

---

**Profa. Elen Macedo Lobato Dra.**  
Coorientadora  
Instituto Federal de Santa Catarina

---

**Prof. Marcos Moecke Dr.**  
Instituto Federal de Santa Catarina

---

**Benoni de Oliveira Pires**  
Instituto Federal de Santa Catarina

São José - SC  
fevereiro/2023

*Este trabalho é dedicado às crianças adultas que, quando pequenas, sonharam em se tornar cientistas.*

# Agradecimentos

Dedico meus sinceros agradecimentos àqueles que acreditaram em minha capacidade, em especial à minha família, aos meus colegas que me deram apoio e a meus orientadores que ajudaram a tornar realidade a chegada deste momento.

*"Na adversidade, uns desistem, outros batem recorde." - Ayrton Senna*

# Resumo

Computação em nuvem (*cloud computing*) é uma arquitetura de computação na internet que vem crescendo em escala e serviços nos últimos anos. O objetivo desta arquitetura é fornecer recursos como processamento e armazenamento de dados sob demanda aos usuários. Esses recursos são uma realidade amplamente utilizada no ambiente corporativo, tendo em vista que eles ajudam a melhorar a agilidade e mobilidade no fornecimento de serviços para os usuários, além de trazer redução de custos com equipamentos e mão-de-obra. Diversas técnicas podem ser utilizadas para melhorar a eficiência no fornecimento dos serviços de computação em nuvem. Por esse motivo, esse tema vem sendo muito discutido em meio acadêmico e corporativo. Com o intuito de simplificar diversos assuntos técnicos que envolvem esta área, este trabalho faz um estudo sobre computação em nuvem, com foco voltado contratação ou implantação de infraestruturas de computação em nuvem por instituições públicas, especialmente órgãos públicos da esfera federal, e mais especificamente ainda, aos institutos federais.

**Palavras-chave:** Computação em nuvem. Arquitetura de rede. Contratações públicas. Legislação.



# Abstract

Cloud computing is an internet computing architecture that has been growing in scale and services in recent years. The purpose of this architecture is to provide features such as on-demand data processing and storage to users. These resources are widely used in the corporate environment, given that they help to improve agility and mobility in providing services to users, in addition to reducing equipment and labor costs. Several techniques can be used to improve the efficiency in providing cloud computing services. For this reason, this theme has been much discussed in academic and corporate circles. In order to simplify several technical matters that involve this area, this work makes a study on cloud computing, with a focus on contracting or implementing cloud computing infrastructures by public institutions, especially public agencies at the federal level, and even more specifically , to federal institutes.

**Keywords:** Cloud Computing. Network architecture. Public Hiring. Legislation.

# Lista de ilustrações

Figura 1 – Modelos de Serviço, segundo <i>National Institute of Standards and Technology</i> (NIST). . . . .	26
Figura 2 – Ciclo DevOps. . . . .	27
Figura 3 – Armazenamento de dados no Ceph. . . . .	31
Figura 4 – Objeto no <i>Ceph File System</i> (CephFS). . . . .	31
Figura 5 – Componentes de um <i>cluster</i> kubernetes. . . . .	33
Figura 6 – Armazenamento com compartilhamento total de disco. . . . .	43
Figura 7 – Armazenamento sem compartilhamento de disco. . . . .	44
Figura 8 – Armazenamento com espelhamento de disco. . . . .	44
Figura 9 – Comparação entre <i>Redundant Array of Inexpensive Drives</i> (RAID)s. . .	45
Figura 10 – Tabela de preços, especificações e quantitativos. . . . .	50
Figura 11 – Tabela de itens, descrição e valores - Contrato 43/2020 da Prefeitura de Pompeia. . . . .	57
Figura 12 – Tabela de itens, descrição e valores - Contrato nº 2022-0015 do TRIBUNAL REGIONAL DO TRABALHO DA 1ª REGIÃO . . . . .	58

# Lista de abreviaturas e siglas

<b>EXT4</b> <i>fourth extended filesystem</i> . . . . .	30
<b>XFS</b> <i>Extended Filesystem</i> . . . . .	30
<b>LAN</b> <i>Local Area Network</i> . . . . .	24
<b>NIST</b> <i>National Institute of Standards and Technology</i> . . . . .	9
<b>CephFS</b> <i>Ceph File System</i> . . . . .	9
<b>ENIAC</b> <i>Electronic Numerical Integrator and Compute</i> . . . . .	20
<b>ARP</b> <i>Ata de Registro de Preços</i> . . . . .	15
<b>CDN</b> <i>Content Delivery Network</i> . . . . .	37
<b>P2P</b> <i>Peer-to-Peer</i> . . . . .	37
<b>IFSC</b> <i>Instituto Federal de Santa Catarina</i> . . . . .	17
<b>IoT</b> <i>Internet of Things</i> . . . . .	59
<b>AOS</b> <i>Arquitetura Orientada a Serviço</i> . . . . .	38
<b>SOA</b> <i>Service-Oriented Architecture</i> . . . . .	38
<b>HPC</b> <i>High Performance Computing</i> . . . . .	37

<b>NLL</b> <i>Nova Lei de Licitações</i> . . . . .	53
<b>ARP</b> <i>Ata de Registro de Preços</i> . . . . .	15
<b>GPU</b> <i>Graphics Processing Unit</i> . . . . .	38
<b>CPU</b> <i>Central Processing Unit</i> . . . . .	26
<b>HD</b> <i>Hard Disk</i> . . . . .	34
<b>RAM</b> <i>Random Access Memory</i> . . . . .	28
<b>SDN</b> <i>Software-defined Networking</i> . . . . .	28
<b>SDDC</b> <i>Software-defined Data Center</i> . . . . .	28
<b>SDS</b> <i>Software-defined Storage</i> . . . . .	28
<b>VM</b> <i>Virtual Machine</i> . . . . .	28
<b>BIOS</b> <i>Basic Input/Output System</i> . . . . .	28
<b>FPGA</b> <i>Field Programmable Gate Array</i> . . . . .	38
<b>TPU</b> <i>Tensor Processing Unit</i> . . . . .	38
<b>API</b> <i>Application Programming Interface</i> . . . . .	30
<b>KVM</b> <i>Kernel Virtual Machine</i> . . . . .	28
<b>OSD</b> <i>Object Storage Daemon</i> . . . . .	30

<b>SaaS</b> <i>Software as a Service</i> . . . . .	25
<b>PaaS</b> <i>Platform as a Service</i> . . . . .	25
<b>IaaS</b> <i>Infrastructure as a Service</i> . . . . .	25
<b>S3</b> <i>Simple Storage Service</i> . . . . .	45
<b>REST</b> <i>Representational State Transfer</i> . . . . .	33
<b>CRUD</b> <i>Create, Read, Update and Delete</i> . . . . .	41
<b>PMem</b> <i>Persistent Memory</i> . . . . .	41
<b>SGBD</b> <i>Sistema de Gerenciamento de Bases de Dados</i> . . . . .	39
<b>ACID</b> <i>Atomicidade, Consistência, Isolamento e Durabilidade</i> . . . . .	39
<b>SQL</b> <i>Structured Query Language</i> . . . . .	39
<b>NoSQL</b> <i>Not only Structured Query Language</i> . . . . .	39
<b>WSDL</b> <i>Web Services Description Language</i> . . . . .	39
<b>UDDI</b> <i>Universal Description, Discovery and Integration</i> . . . . .	39
<b>XML</b> <i>Extensible Markup Language</i> . . . . .	39
<b>REDIS</b> <i>Remote Dictionary Server</i> . . . . .	40
<b>WSL</b> <i>Windows Subsystem For Linux</i> . . . . .	40

<b>EEPROM</b> <i>Electrically-Erasable Programmable Read-Only Memory</i> . . . . .	43
<b>SSD</b> <i>Solid-State Drive</i> . . . . .	43
<b>RAID</b> <i>Redundant Array of Inexpensive Drives</i> . . . . .	9
<b>EGD</b> <i>Estratégia de Governança Digital</i> . . . . .	47
<b>TIC</b> <i>Tecnologia de Informação e Comunicação</i> . . . . .	17
<b>PDTIC</b> <i>Plano Diretor de Tecnologia da Informação e Comunicação</i> . . . . .	17
<b>ETP</b> <i>Estudo Técnico Preliminar</i> . . . . .	50
<b>NLL</b> <i>Nova Lei de Licitações</i> . . . . .	53
<b>LOA</b> <i>Lei Orçamentária Anual</i> . . . . .	52
<b>TCU</b> <i>Tribunal de Contas da União</i> . . . . .	54
<b>SAEB</b> <i>Secretaria de Administração do Governo do Estado da Bahia</i> . . . . .	54
<b>PNCP</b> <i>Portal Nacional de Contratações Públicas</i> . . . . .	54
<b>CAU/SC</b> <i>Conselho de Arquitetura e Urbanismo de Santa Catarina</i> . . . . .	56
<b>SERPRO</b> <i>Serviço Federal de Processamento de Dados</i> . . . . .	56
<b>TRT1</b> <i>Tribunal Regional do Trabalho da 1ª Região - RJ</i> . . . . .	58
<b>HTTP</b> <i>Hypertext Transfer Protocol</i> . . . . .	42

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
<b>1.1</b>	<b>Motivação</b>	<b>17</b>
<b>1.2</b>	<b>Objetivos</b>	<b>18</b>
1.2.1	Objetivo principal	18
1.2.2	Objetivos específicos	19
<b>1.3</b>	<b>Organização do texto</b>	<b>19</b>
<b>2</b>	<b>COMPUTAÇÃO EM NUVEM</b>	<b>20</b>
<b>2.1</b>	<b>Abordagem histórica</b>	<b>20</b>
<b>2.2</b>	<b>Tipos de arquitetura de sistemas computacionais</b>	<b>21</b>
<b>2.3</b>	<b>Principais tecnologias de computação em nuvem</b>	<b>21</b>
2.3.1	Computação distribuída	22
2.3.2	Agregado	23
2.3.3	Grade computacional	23
<b>2.4</b>	<b>O que é computação em nuvem</b>	<b>24</b>
2.4.1	Características essenciais	24
2.4.2	Modelos de serviço	25
<b>2.5</b>	<b>DevOps</b>	<b>27</b>
<b>2.6</b>	<b>Virtualização</b>	<b>28</b>
2.6.1	Máquinas virtuais	28
2.6.2	Contêiner	28
2.6.3	VM completa	29
2.6.4	VM plena	30
2.6.5	<i>Software-defined as a data center</i>	30
2.6.6	<i>Software-defined Storage</i>	30
2.6.7	Ceph	30
2.6.8	Rook	31
2.6.9	Kubernetes	32
<b>2.7</b>	<b>Escalabilidade</b>	<b>34</b>
<b>2.8</b>	<b>Pagamento pontual (PAY-AS-YOU-GO)</b>	<b>35</b>
<b>3</b>	<b>PRINCÍPIOS DA COMPUTAÇÃO EM NUVEM</b>	<b>37</b>
<b>3.1</b>	<b>Computação como utilidade</b>	<b>37</b>
<b>3.2</b>	<b><i>High Performance Computing (HPC)</i></b>	<b>37</b>
<b>3.3</b>	<b>Arquitetura Orientada a Serviços (AOS)</b>	<b>38</b>
<b>3.4</b>	<b>Banco de dados</b>	<b>39</b>

3.4.1	Banco de dados relacional . . . . .	40
3.4.2	Banco de dados não relacional . . . . .	40
3.4.3	Persistência de dados . . . . .	41
<b>3.5</b>	<b><i>Content Delivery Network (CDN)</i></b> . . . . .	<b>46</b>
<b>3.6</b>	<b><i>Peer to peer (P2P)</i></b> . . . . .	<b>46</b>
<b>3.7</b>	<b>CDN-P2P</b> . . . . .	<b>46</b>
<b>4</b>	<b>CONTRATAÇÃO DE SERVIÇO DE NUVEM EM ÓRGÃOS PÚBLI- COS</b> . . . . .	<b>47</b>
<b>4.1</b>	<b>Legislação</b> . . . . .	<b>47</b>
4.1.1	Documentos norteadores do IFSC . . . . .	48
4.1.2	Serviços disponíveis na Ata de Registro de Preços (ARP) 11/2021 . . . . .	49
<b>4.2</b>	<b>Processo de contratação pública</b> . . . . .	<b>50</b>
4.2.1	Planejamento . . . . .	52
4.2.2	Execução . . . . .	54
4.2.3	Contratação . . . . .	55
<b>4.3</b>	<b>Tópicos relevantes para implantação/contratação de serviço em nuvem</b> . . . . .	<b>56</b>
4.3.1	Contratações de serviço em nuvem em outros órgãos públicos . . . . .	56
<b>4.4</b>	<b>Equipe de profissionais na área disponíveis</b> . . . . .	<b>59</b>
<b>4.5</b>	<b>Infraestrutura própria <i>versus</i> alugada</b> . . . . .	<b>60</b>
4.5.1	Contratação padronizada . . . . .	61
4.5.2	Contratação personalizada . . . . .	61
<b>5</b>	<b>CONCLUSÕES</b> . . . . .	<b>63</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>65</b>



# 1 Introdução

Nos dias atuais, empresas buscam cada vez mais meios para facilitar a prestação de serviços e simplificar a vida de seus clientes e seus funcionários. Uma maneira de fazer isso é através de prestação de serviços através da internet, sejam eles *Websites*, páginas de cadastro, sistemas de gestão, jogos online, fornecimento de arquivos, entre outros. Esse tipo de serviço tem ganhado força nos últimos anos pelo fato de evitar deslocamentos desnecessários até um estabelecimento para conhecer seus produtos e serviços e também pela segurança de manter as informações armazenadas na nuvem, reduzindo a possibilidade de perdas por falhas em componentes físicos dos provedores de serviços, seja por algum dano físico sofrido por componentes de forma acidental ou pelo desgaste natural dos componentes. Fato é que muitas empresas passaram a prover seus serviços na nuvem. Isso fez com que inúmeras pesquisas fossem realizadas, tanto por parte das empresas e seus profissionais contratados quanto em nível acadêmico [ARAÚJO \(2017\)](#).

Dentro do ambiente acadêmico, e podemos tomar como exemplo o próprio curso de Engenharia de Telecomunicações do Instituto Federal de Santa Catarina, diversos artigos e trabalhos de conclusão de curso abordam como tema o assunto computação em nuvem, ou *cloud computing* [SOUZA \(2014a\)](#). Esse é o termo coloquial que passou a ser utilizado quando nos referimos ao fornecimento de serviços pela internet a que se refere o parágrafo anterior. Ocorre que, normalmente, outros trabalhos utilizam como meio para chegar ao objetivo do trabalho a computação em nuvem, mas acaba que sua finalidade é voltada para o uso de alguma outra tecnologia específica. Em resumo, há todo um embasamento teórico necessário referente ao tema computação em nuvem que é comum para a maioria destes trabalhos. Por isso, a ideia deste trabalho é fornecer um embasamento teórico para suprir as demandas sobre computação em nuvem voltado para justificar e orientar uma tomada de decisão para a contratação deste tipo de serviço em uma determinada instituição.

Normalmente no serviço público, funcionários públicos possuem conhecimento técnico em suas áreas de atuação. Contudo, muitas vezes ocorre de eles serem nomeados para cargos de gestão, ou simplesmente necessitam elaborar documentos necessários para realizar compras ou contratações na organização a qual fazem parte. Como o processo para compra ou contratação no serviço público é bastante burocrático e necessita diversos cuidados, para que a contratação atenda corretamente aos requisitos da contratante e para que nenhuma regra seja descumprida, este trabalho se propõe a fornecer suporte para o servidor público que necessite realizar uma aquisição de serviços de computação em nuvem. Este guia inclui, além de um resumo bibliográfico sobre os assuntos mais relevantes envolvendo computação em nuvem, de maneira resumida e de fácil compreensão por alguém que desconheça do assunto, mas também um resumo sobre como funcionam os

processos de contratação em órgãos públicos, alguns exemplos de contratação de serviço de computação em nuvem por outras instituições.

Outro ponto a ser abordado neste trabalho é referente a contratações públicas de serviços de computação em nuvem, com um foco voltado para a legislação do Instituto Federal de Santa Catarina. Porém, a maioria dos conceitos sobre computação em nuvem e sobre processos de contratação pública podem ser aplicados a outras instituições a que se deseje implementar tais serviços.

## 1.1 Motivação

Em praticamente todas as áreas econômicas, redução de custos é algo normalmente levada em consideração. A busca por soluções que consigam reunir tanto eficiência quanto baixo custo cada vez mais têm sido pauta, seja no setor privado quanto no setor público. O Plano Diretor de Tecnologia da Informação e Comunicação ([PDTIC](#)), aprovado pela RESOLUÇÃO CGD Nº 04, DE 09 DE ABRIL DE 2021, o qual traz as diretrizes a serem seguidas na área de Tecnologia da Informação para o Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina para a gestão nos anos de 2021 e 2022. Nesse documento consta o RELATÓRIO DE RESULTADOS DO [PDTIC](#) 2018 – 2020. Esse relatório aponta que de 3 (três) ações do projeto de redução de infraestrutura de data center nos Câmpus, 2 (duas) delas ainda não foram concluídas. São elas:

- Ação 1: Centralização de infraestrutura e serviços no data center da reitoria. (NÃO CONCLUÍDO).
- Ação 2: Utilização de serviços disponíveis pela Rede Nacional de Ensino e Pesquisa. (CONCLUÍDO).
- Ação 3: Contratação de serviço em nuvem. (NÃO CONCLUÍDO).

Ou seja, de acordo com a Ação 1, haverá centralização em data centers localizados na Reitoria do Instituto Federal de Santa Catarina ([IFSC](#)). E a ação 3 orienta a contratação de serviços em nuvem. Além disso, outros projetos de melhoria na eficiência de gastos com Tecnologia de Informação e Comunicação ([TIC](#)) e ampliação de enlaces de rede nos Câmpus do interior já foram concluídos ou estão em andamento.

Pois bem, temos o seguinte cenário: centralização de data center na Reitoria; necessidade de redução de custos com infraestrutura; enlaces de rede adequados disponíveis em praticamente todos os Câmpus; e orientação para utilização de serviços na nuvem. Tudo isso mostra a necessidade de haver um trabalho voltado para o estudo de tecnologias de computação em nuvem e que poderão servir de base para a escolha de uma, ou de um conjunto delas.

Há outros trabalhos de conclusão de curso que abordam o tema computação em nuvem, porém, há pouca teoria sobre o assunto. Outros cursos de graduação ofertados pelo IFSC, além da Engenharia de Telecomunicações (Câmpus São José), possuem grades que podem abordar o tema, como os cursos de Análise e Desenvolvimento de Sistemas (Câmpus Tubarão, Gaspar e Canoinhas), Gestão de Tecnologia da Informação (Câmpus Florianópolis), Sistemas para Internet (Câmpus Garopaba), Ciências da Computação (Câmpus Lages) e Sistemas da Informação (Câmpus Caçador). Portanto, há demanda por referências no assunto que possa ser utilizado em todos esses locais.

Como há pouca referência acadêmica, setores de tecnologia da informação e gestores possuem pouco material de referência para ser utilizado na implantação deste tipo de serviço, sejam eles dentro ou fora da instituição do IFSC. Por este motivo, há uma demanda por um trabalho que possa ser utilizado como fonte documentada sobre o tema, quais seus tipos, tecnologias envolvidas e particularidades envolvendo a contratação ou implantação deste tipo de serviços.

Portanto, tendo como justificativa o cenário relatado anteriormente nesta seção, este trabalho traz um estudo sobre tecnologias em nuvem para fornecimento de conteúdo para usuários. As principais características dessas tecnologias relatadas neste trabalho são a baixa necessidade de investimento com data centers para replicar informações, necessidade de enlaces de rede, alta disponibilidade de conteúdo em regiões geograficamente próximas e escalabilidade. Com isso, será possível a utilização destas tecnologias para resolver a demanda de serviços apontados anteriormente no PDTIC, além de nortear outros trabalhos que por ventura possam utilizar assuntos envolvidos com *cloud computing*.

Uma outra justificativa para este trabalho é fornecer a gestores públicos e demais interessados um documento de fácil compreensão, inclusive para aqueles que desconhecem sobre o assunto. Com isso, traçando os avanços da computação em nuvem, analisando legislações sobre contratações públicas e em especial algumas normas internas do Instituto Federal de Santa Catarina sobre o assunto, este trabalho poderá subsidiar tomadas de decisões de equipes de TI (Tecnologia da Informação) para a implantação e contratação de serviços em nuvem.

## 1.2 Objetivos

Os objetivos deste trabalho estão descritos abaixo:

### 1.2.1 Objetivo principal

O objetivo principal deste trabalho é fornecer a profissionais que atuam na área de computação em nuvem em instituições públicas um tutorial sobre como realizar contratações na área, visando buscar maior eficiência nos serviços prestados.

### 1.2.2 Objetivos específicos

Como objetivos específicos deste trabalho tem-se a criação uma base documental sobre computação em nuvem, abordando os principais fundamentos, tecnologias e temas necessários para tornar simples a compreensão sobre este assunto.

Outro objetivo deste trabalho é informar ao leitor, de forma resumida e de simples compreensão, como funcionam os processos de contratação de serviços e aquisição de produtos em órgãos públicos.

Um foco especial será dado para possibilidades de contratações de serviços em nuvem pelo Instituto Federal de Santa Catarina, tendo como base as normas internas desta instituição.

## 1.3 Organização do texto

Este trabalho está organizado com a seguinte estrutura:

O capítulo 1 traz a introdução do trabalho, sua motivação, objetivos e organização do texto.

Os capítulos 2 e 3 trazem os fundamentos teóricos necessários para a compreensão do tema "Computação em Nuvem". Já o capítulo 4 aborda formas de contratação de serviço de computação em nuvem por órgãos públicos, em especial, pelo [IFSC](#), tendo em vista o cenário atual da infraestrutura de rede, a redução de investimentos com equipamentos e demais orientações apontadas pelo [PDTIC](#) da gestão de 2021/2022. Por último, o capítulo 5 traz as considerações finais sobre o assunto computação em nuvem e as principais vantagens e desvantagens de sua implantação para melhoria nos serviços de entrega de conteúdo.

## 2 Computação em nuvem

Neste capítulo iremos primeiramente fazer um breve apanhado histórico sobre os primórdios da computação em nuvem. Em seguida, algumas definições importantes sobre o assunto, como os tipos de arquitetura envolvendo sistemas computacionais e a definição do NIST para computação em nuvem. Assuntos referentes a computação distribuída, que estão diretamente relacionados com a este tema, serão abordados neste capítulo, bem como outros assuntos referentes a virtualização, serviços ou plataformas de computação em nuvem, como Ceph, Rook e Kubernetes.

### 2.1 Abordagem histórica

Em um apanhado histórico, podemos começar a falar de computação em nuvem com a criação do *Electronic Numerical Integrator and Computer* ([ENIAC](#)), que foi o primeiro computador digital eletrônico criado na década de 1940. Podemos considerar que este foi o primeiro *mainframe* a existir.

Segundo [Botole \(2011\)](#), *mainframe* é um servidor de grande porte que é utilizado para armazenar bases de dados, servidores de transações e aplicações que requerem grande capacidade de processamento, segurança e disponibilidade. É um repositório central de dados em uma central de processamento corporativa, ligada aos usuários através de dispositivos de menor capacidade de processamento, como terminais ou estações de trabalho. A presença de um mainframe frequentemente implica em uma computação centralizada, diferentemente da computação distribuída, que será abordada em seguida.

Segundo [Botole \(2011\)](#), existiram até hoje, três gerações de mainframes. A primeira geração era baseada em válvulas, ou tubos. A segunda geração era baseada nos transistores. E a terceira e atual geração baseada nos circuitos integrados. O primeiro *mainframe* de terceira e atual geração foi o System/360, que foi o primeiro mainframe da família S lançado pela IBM, em 1964.

O System/360 foi importante, pois a partir dele, *hardware* e *software* passaram a ser padronizados e oferecido para os clientes, tornando possível a portabilidade das aplicações. A partir daqui, surge o conceito de arquitetura de *software*, que segundo [VIANA \(2018\)](#), pode ser definida como sendo um conjunto de estruturas necessárias de um sistema, compreendendo elementos de *software*, relacionamentos entre eles e as propriedades de ambos. Estas estruturas ou camadas, tratam-se de divisões lógicas em uma aplicação, ou ainda, a maneira como o código da aplicação é dividido, com a intenção de reduzir o acoplamento entre partes da aplicação que possuem funções distintas. Existem diversos

modelos de camadas para aplicações, contudo, aquelas que apresentam apenas uma camada são muito comuns nos *mainframes*, os quais centralizavam todo o processo. Utilizar apenas uma camada traz a vantagem de reduzir o tempo de resposta, pois não é necessário fazer consultas na rede. Porém, a escalabilidade pode ser comprometida com o aumento no número de usuários, pois dessa forma, a aplicação não pode ser distribuída separadamente.

Para os modelos com apenas uma camada, existem duas maneiras de escalonar um servidor, o escalonamento vertical e o escalonamento horizontal.

## 2.2 Tipos de arquitetura de sistemas computacionais

Em um sistema computacional, existem 3 (três) tipos de arquiteturas, a saber: a arquitetura centralizada, descentralizada e híbrida. Em uma arquitetura centralizada um único computador centraliza todo o processamento, armazenamento e acesso ao sistema. Sistemas centralizados seguem um modelo cliente-servidor, onde todos os clientes acessam os recursos de um servidor. Os *mainframes* utilizam este tipo de abordagem. Sua principal vantagem é que os sistemas são mais simples e não existem problemas de sincronização e consolidação. As desvantagens deste tipo de abordagem são a baixa escalabilidade e baixa tolerância a falhas. Outra desvantagem a se considerar, é que neste tipo de arquitetura, atrasos na rede ou de processamento podem ocorrer com maior frequência, pois apenas o servidor fica responsável por atender todo o processamento.

Uma arquitetura descentralizada de sistemas computacionais consiste em realizar todo o processamento de forma distribuída, sem a existência de um servidor que atua em toda a comunicação. As vantagens deste tipo de arquitetura é que ela é mais tolerante a falhas, pois o conteúdo não está localizado no servidor central, mas sim em outros dispositivos espalhados pela rede. Outra vantagem desta arquitetura é que ela é mais escalável, pois a comunicação não acontece necessariamente para um dispositivo centralizado (servidor), ela pode ocorrer entre quaisquer pares. A desvantagem deste tipo de arquitetura é que realizar a sincronização dos dados entre os pares e manter a consistência dos dados quando alguma cópia é alterada é uma tarefa complexa.

Já uma arquitetura híbrida combina as características de uma arquitetura centralizada e descentralizada.

## 2.3 Principais tecnologias de computação em nuvem

Nesta sessão, serão abordados alguns assuntos importantes para a compreensão de como funcionam as tecnologias utilizadas para computação em nuvem. Algumas definições e as tecnologias mais importantes sobre o assunto também serão apresentadas.

### 2.3.1 Computação distribuída

Segundo [TANENBAUM e STEEN \(2007\)](#), um sistema distribuído se caracteriza por ser um conjunto de computadores independentes, porém, interconectados por uma rede e que trabalha de forma cooperativa, com um objetivo específico e apresentam ao usuário a sensação de estar utilizando um único sistema sem que os mesmos percebam estar utilizando um sistema distribuído.

Os autores [DEITEL, DEITEL e CHOFFNES \(2005\)](#) afirmam que sistemas distribuídos podem conter computadores que não necessariamente estão juntos no mesmo ambiente, ou seja, se distribuem geograficamente podendo estar em outros continentes, por exemplo.

Segundo [Silva \(2012\)](#), com o desenvolvimento de tecnologias nas áreas de computação distribuída, é possível utilizar um conjunto de recursos, de maneira coordenada, tais como processamento, armazenamento e memória, para assim resolver problemas que necessitam de alta demanda de recursos computacionais. Dessa forma, o recurso necessário pode ser obtido através da união de esforços de dois ou mais computadores conectados a uma mesma rede, fazendo com que a informação seja processada com maior eficiência.

Existem três formas de armazenar as informações em um sistema: local, remoto ou distribuído. No armazenamento local, as informações são armazenadas no disco rígido localizado na mesma máquina da aplicação. Quando o armazenamento ocorre na forma remota, as informações são armazenadas em um disco rígido que encontra-se em um local remoto onde está rodando a aplicação. Já em um armazenamento distribuído, a informação é armazenada em vários servidores, que atuam de maneira coordenada e totalmente transparente para o usuário, com o intuito de prover redundância e melhorar a disponibilidade dos arquivos.

Quando falamos em computação distribuída, [DEITEL, DEITEL e CHOFFNES \(2005\)](#) traz uma reflexão importante a ser feita sobre o conceito de paralelismo. Uma das formas de realizar o paralelismo consiste em implementar técnicas como multiprocessamento, multiprogramação e multithread, todas elas **localmente**, em um mesmo computador, para executar tarefas semelhantes, em paralelo, e melhorar o desempenho de determinada requisição.

Uma forma diferente de se realizar tarefas de forma paralela é fazendo isso através de uma rede de computadores, entre várias máquinas. Ou seja, vários computadores conectados a uma mesma rede realizam tarefas semelhantes, de forma paralela, com o intuito de ganhar desempenho e escalabilidade. Dessa forma, é possível criar uma grande rede de computadores interligados, distribuídos fisicamente ao redor do mundo, onde qualquer um deles (ou até todos) pode tratar requisições vindas de um usuário/cliente e reduzir o tempo necessário para executar esta tarefa. Outra vantagem desse tipo de

abordagem em paralelo é que, caso necessário, é possível adicionar computadores na rede para melhorar o desempenho, na quantidade que se desejar. Coisa que no paralelismo realizado localmente (em apenas um computador) haveria um limite físico caso se desejasse aumentar o *hardware* de um computador para melhorar o processamento.

Deve haver algum mecanismo para possibilitar a comunicação entre um usuário (cliente) e um sistema distribuído com vários computadores. Uma maneira de realizar essa comunicação é através de trocas de mensagens. Podemos classificar as de trocas de mensagem quanto a forma de comunicação e forma de sincronização. A forma de comunicação pode ser direta ou indireta. Na forma de comunicação direta, as mensagens entre emissor e receptor são enviadas utilizando um número de processo conhecido. A comunicação se dá diretamente entre o emissor e o receptor em questão. Por outro lado, na forma de comunicação indireta, o emissor envia uma mensagem para uma porta de comunicação, mas não tem o conhecimento de qual receptor irá receber sua requisição. O receptor apenas responde a requisição, sem saber qual sua origem.

Referente forma de sincronização na troca de mensagens, ela pode ser síncrona ou assíncrona. Na síncrona (bloqueante), o emissor espera até que receptor receba a mensagem, assim como o receptor aguarda até que a mensagem esteja disponível para o emissor. Já na forma assíncrona (não bloqueante), o emissor envia a mensagem e não aguarda até que a mesma seja recebida pelo receptor. Receptor recebe a mensagem se ela estiver disponível, caso contrário, informa para o emissor que a mensagem não estava disponível.

### 2.3.2 Agregado

Agregado, também conhecido como *cluster* é um agrupamento de vários processadores com configuração paralela para que estes executem uma tarefa específica, para que esta seja executada em um tempo menor. Geralmente, os computadores que formam um *cluster* são heterogêneos e estão interligados através de uma rede de alta velocidade. Quando um *cluster* é criado, seus objetivos são os de obter alto desempenho no processamento de uma tarefa específica, escalabilidade e alta disponibilidade de um determinado conteúdo (RAVANÊDA, 2014).

### 2.3.3 Grade computacional

Computação em grade é quando agregamos vários computadores na rede, formando assim um sistema distribuído, com o intuito de resolver tarefas mais complexas. Normalmente, um sistema em grade é utilizado para implementar os sistemas de controle em um sistema distribuído, através de mecanismos de balanceamento de carga, concorrência, comunicação de processos entre máquinas, escalabilidade e tolerância a falhas.

As características de computadores em grade são semelhantes ao de um agregado.



Porém, a principal diferença entre ambos é que em um agregado, as configurações de *hardware* e sistema operacional são idênticas. Diferente de uma grade computacional, onde qualquer computador, com quaisquer configurações de *hardware* e *software* podem trabalhar em conjunto para resolver problemas. Estes computadores sequer precisam estar conectados através da mesma *Local Area Network* (LAN).

Entretanto, é possível instalar *softwares* nos computadores em grade e *cluster*, e então agregar vários *clusters* e grades. Isso faz com que seja possível obter as vantagens de ambos os cenários e maximizar ainda mais o desempenho. Um termo novo tem sido utilizado para este tipo de situação, onde vários *clusters* e grades são agregados para unir seus poderes de processamento. Esse termo é o *cloud computing*, ou em português, computação em nuvem.

## 2.4 O que é computação em nuvem

O termo computação em nuvem (do inglês *cloud computing*) é uma metáfora utilizada para representar a internet e a infraestrutura de comunicação entre arquiteturas. Ela representa uma abstração que oculta toda a complexidade da infraestrutura.

Computação em nuvem também pode ser definida como a capacidade de acessar recursos de qualquer dispositivo na internet, ou ainda, é a entrega de soluções como serviço através da nuvem.

O NIST é uma agência governamental não regulatória da administração de tecnologia do Departamento de Comércio dos Estados Unidos. Para definir uma linha de base de discussão sobre o que é computação na nuvem, o NIST criou uma definição sobre o tema, o qual caracteriza aspectos importantes, conforme segue.

Segundo NIST (2011), a definição de computação em nuvem é a seguinte:

A computação em nuvem é um modelo para permitir o acesso onipresente, conveniente e sob demanda à rede compartilhada a um conjunto de recursos de computação configuráveis (por exemplo, redes, servidores, armazenamento, aplicativos e serviços) que podem ser rapidamente provisionado e liberado com o mínimo de esforço de gerenciamento ou interação do provedor de serviços. Este modelo de nuvem é composto por cinco características essenciais, três modelos de serviço e quatro modelos de implantação. (tradução nossa)

### 2.4.1 Características essenciais

As características essenciais citadas acima, segundo NIST (2011), são descritas a seguir:

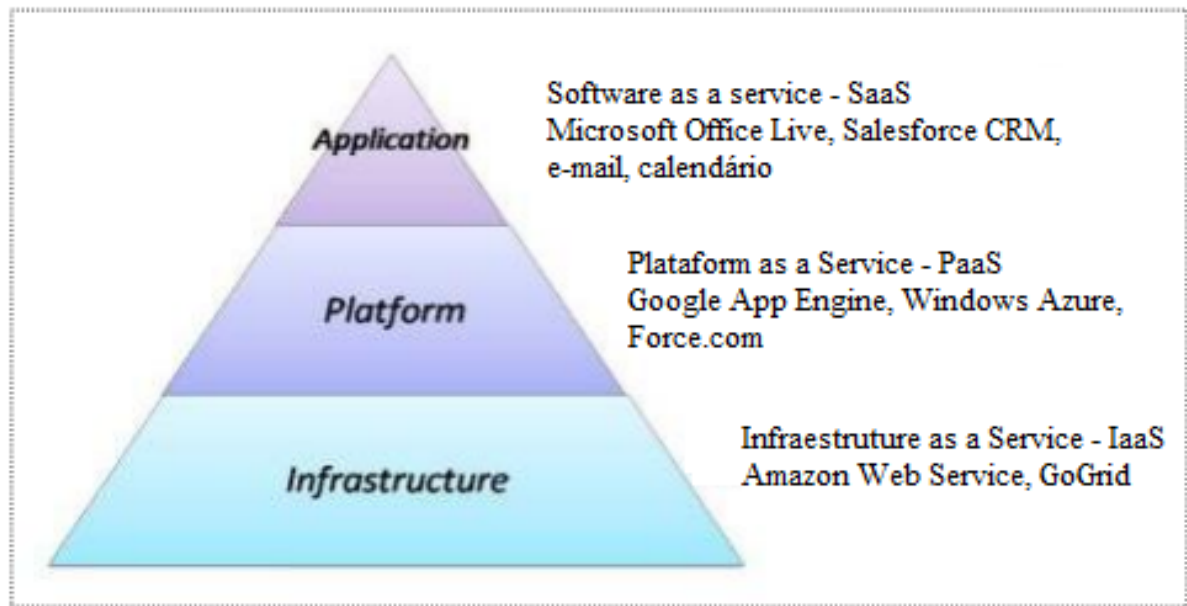
- Auto-serviço sob demanda: é possível fornecer recursos computacionais, como tempo de servidor e armazenamento de rede, conforme a necessidade do consumidor. Tudo isso de forma automática e sem a necessidade de interações humanas com os provedores de serviço.
- Amplo acesso a rede: os recursos ficam disponíveis na rede e podem ser utilizados por usuários em plataformas heterogêneas (celulares, tablets, computadores, etc).
- Agrupamento de recursos: os recursos computacionais são fornecidos para atender a vários consumidores. Esses recursos incluem armazenamento, processamento, memória e largura de banda da rede e eles são alocados dinamicamente, conforme a demanda.
- Rápida elasticidade: os recursos podem ser liberados e providos de forma automática e elástica, dando a impressão ao consumidor que eles são ilimitados e que podem ser adquiridos a qualquer momento e em qualquer quantidade.
- Serviço sob medida: os sistemas em nuvem devem ser controlados e otimizados automaticamente. O uso dos recursos (armazenamento, processamento e largura de banda) pode ser monitorado, relatado e controlado, de maneira transparente, tanto para o provedor quanto para o consumidor.

### 2.4.2 Modelos de serviço

Os principais modelos de serviço citados acima, segundo [NIST \(2011\)](#), podem ser exemplificados na figura 1, abaixo:

- *Software as a Service (SaaS)*: é um modelo de aplicação em nuvem hospedada em um provedor de serviços e então, acessada pelo usuário através de um navegador ou interface de programa. O consumidor não controla recursos da infraestrutura da nuvem subjacente, tais como rede, servidores, sistemas operacionais, armazenamento ou recursos de aplicativos individuais. O acesso aos serviços devem ser totalmente transparentes para o usuário, e para isso, um [SaaS](#) utiliza-se de *Platform as a Service (PaaS)* e *Infrastructure as a Service (IaaS)* para prover seus serviços.
- Plataforma como um serviço ([PaaS](#)): neste caso, o serviço fornecido é um ambiente contendo os recursos necessários para se desenvolver aplicações para a internet, sem a necessidade de saber qual a quantidade de processadores e memória que a aplicação irá utilizar. Uma [PaaS](#) tem como vantagem a automatização de processos de implantação de aplicações para infraestrutura. Uma [PaaS](#) utiliza de uma [IaaS](#) para fornecer a infraestrutura com compatibilidade na linguagem de programação, sistema operacional e ambiente de desenvolvimento.

Figura 1 – Modelos de Serviço, segundo NIST.



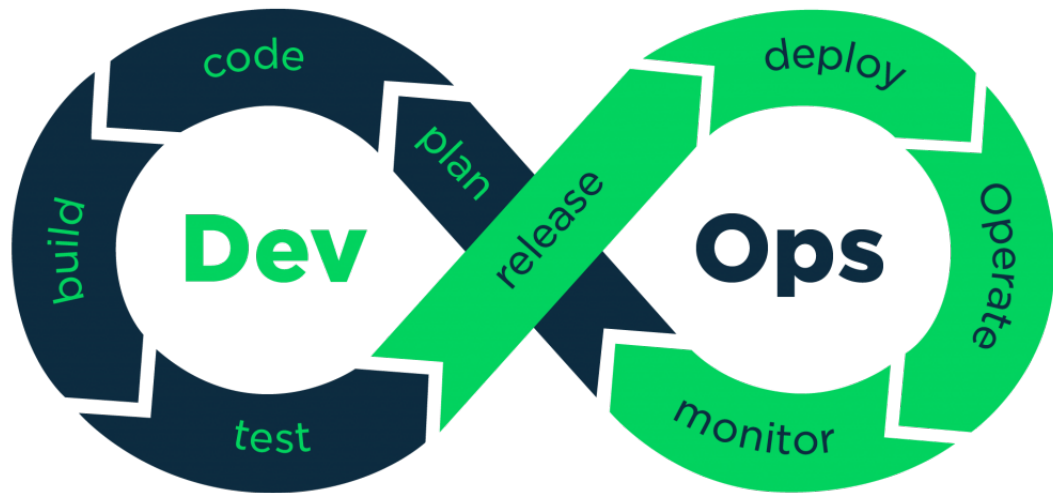
Fonte: SOUSA (2014a).

- Infraestrutura como um serviço (IaaS): este tipo de serviço aluga para os consumidores recursos de infraestrutura em formato de serviço. Estes recursos podem ser espaço em disco, ciclos de *Central Processing Unit* (CPU), memória, recursos de rede, etc. O locador pode ajustar a infraestrutura de acordo com sua necessidade. Normalmente, este serviço possui uma interface que possibilite ao locador administrar sua infraestrutura alugada, além de promover a comunicação dos *hosts* com os demais equipamentos da infraestrutura, inclusive dando suporte para a inclusão de novos equipamentos. Este tipo de serviço provê a infraestrutura das camadas intermediária e superior.

NIST (2011) também define os modelos de implantação em computação em nuvem. São eles:

- Nuvem privada - A infraestrutura é de uma organização. Os recursos podem ser administrados pela própria organização, por terceiros ou ambos.
- Nuvem comunitária - Infraestrutura dividida entre organizações com interesses em comum. Recursos podem ser administrados pelas organizações ou por terceiros.
- Nuvem pública - Infraestrutura está disponível para o público em geral. Os recursos são administrados e organizados por quem vende o serviço de computação em nuvem.
- Nuvem híbrida - União de uma ou mais nuvens (privada, comunitária ou pública).

Figura 2 – Ciclo DevOps.



Fonte: [Leone \(2022\)](#)

## 2.5 DevOps

O termo DevOps surgiu da contração de duas palavras do inglês, *development* e *operations*. Ela representa um movimento cultural e profissional na área de engenharia de software. Seu objetivo é aproximar as equipes de desenvolvimento e de infraestrutura. Normalmente, cada equipe possuía as próprias metas para obter melhores resultados. Mas com o surgimento dessa nova cultura, ambas as equipes começaram a trabalhar juntas no mesmo projeto e compartilhar as mesmas metas. Segundo [BUFFA \(2021\)](#), uma proposta do DevOps é contratar "operadores que pensam como desenvolvedores" e "desenvolvedores que pensam como operadores". Este último autor também traz todo um apanhado histórico da criação do termo, no capítulo 2.1 de seu trabalho.

Dentro da cultura DevOps, existe o processo DevOps, que consiste em oito etapas, divididas em dois grupos, e todas elas interligadas, formando um símbolo de infinito, para representar um ciclo que nunca se acaba, como mostra a figura 2.

Sendo assim, as etapas de desenvolvimento (à esquerda) são *code* (codificação), *build* (lançamento de *builds*) e *test* (testes). Já as etapas de operação (à direita) são *deploy* (implantação), *operate* (operação) e *monitor* (monitoramento). Através das etapas de *plan* (planejamento) e *release* (liberação) ocorre a integração entre *dev* (desenvolvimento) e *ops* (operação). Dessa forma, o ciclo infinito se fecha no desenvolvimento e operação de um sistema em nuvem.

Segundo [Gaea Consulting \(2022\)](#), os 5 (cinco) benefícios da cultura DevOps são:

- Segurança;

- Agilidade nas entregas;
- Aumento da confiabilidade;
- Colaboração entre equipes;
- Redução de custos.

## 2.6 Virtualização

Virtualização é uma abstração dos recursos de *hardware* de uma máquina. Com isso, é possível transformar apenas uma máquina “real” em outras infinitas máquinas “virtuais” e lógicas. Com a virtualização é possível executar qualquer sistema operacional ou qualquer aplicativos. A virtualização mais conhecida é aquela em que se criam máquinas virtuais, ou *Virtual Machine* (VM).

Outras formas de virtualização são *Software-defined Networking* (SDN), *Software-defined Data Center* (SDDC) e *Software-defined Storage* (SDS), as quais serão abordadas com mais detalhes a seguir.

### 2.6.1 Máquinas virtuais

VM consiste em uma máquina virtual que simula um computador virtual e fornece recursos como CPU, *Random Access Memory* (RAM), drives, placas de rede, *Basic Input/Output System* (BIOS) e etc, assim o usuário pode gerenciar a quantidade de máquinas virtuais ou recursos que a aplicação necessite.

Quando se fala em máquinas virtuais, algo importante a ser mencionado é sobre o hipervisor, que trata-se de um software que cria e executa as máquinas virtuais. Com ele, é possível gerenciar os recursos de uma máquina virtual e gerenciá-los em relação aos recursos físicos de uma máquina. Com o hipervisor, é possível que vários sistemas operacionais compartilhem os mesmos recursos de *hardware* que foram virtualizados. Sem ele, isso não seria possível.

Existem algumas opções de hipervisores, que podemos classificar em 2 (dois) grupos: código aberto (*open source*), por exemplo o *Kernel Virtual Machine* (KVM), que faz parte do kernel do linux, além do Xen e código fechado, por exemplo o VMware que é muito popular, e também o Microsoft Hyper-V.

### 2.6.2 Contêiner

Em computação em nuvem, um contêiner consiste em um código de um aplicativo e seus arquivos de configuração, bem como as bibliotecas necessários para a execução

deste aplicativo. Com ele os desenvolvedores e os profissionais de TI podem implantar aplicativos diretamente nos ambientes.

Atualmente, o termo contêineres é comumente associado ao produto da empresa Docker ([DOCKER, 2022](#)). Contêineres são muito similares a máquinas virtuais, porém, com uma vantagem: eles são mais portáteis e mais eficientes não havendo necessidade de se utilizar um sistema operacional visitante, bem como toda a memória necessária para a execução deste último. Também é possível utilizar contêineres dentro de máquinas virtuais, para que assim seja possível uma máquina virtual gerenciar uma quantidade maior de recursos com uma menor quantidade de máquinas virtuais criadas. Isso tudo sem deixar de garantir o isolamento entre processos. Atualmente, existem plataformas de orquestração de contêineres. A finalidade de se orquestrar contêineres é ter uma plataforma onde seja possível gerenciar e coordenar de maneira simples e eficiente, diversos recursos computacionais independentes. Podemos citar algumas plataformas de orquestração de contêineres famosas, como Google Kubernetes, Apache Mesos e Docker Swarm.

### 2.6.3 VM completa

A virtualização completa ou virtualização total consiste em simular um computador virtual, fornecendo recursos virtuais como [CPU](#), [RAM](#), drives, placas de rede, [BIOS](#) e etc. Com o acesso a estes recursos, é possível obter quase a mesma velocidade de processamento dos sistemas hóspedes como se não houvesse virtualização [SOUSA \(2014a\)](#).

Na técnica de virtualização completa, cada acesso de entrada e saída que o usuário executa irá gerar um *trap*, que é controlado pelo monitor de máquinas virtuais. Através do tipo de entrada/saída que o usuário realizar no sistema operacional com este tipo de virtualização, o monitor de máquinas virtuais consegue emular uma cópia do *hardware* e então retorná-la para o sistema convidado.

O principal problema deste tipo de técnica é que para a sua utilização, o monitor de máquinas virtuais e o sistema convidado precisam se comunicar em linguagem de protocolo de *hardware*. Este tipo de comunicação é muito eficiente para acessar o *hardware* de uma máquina real. Porém, entre softwares isso é muito lento e ineficiente. Contudo, alguns fabricantes deste tipo de solução desenvolvem *drivers* especiais que utilizam recursos de entrada e saída da máquina real. Com a utilização destes recursos da máquina real, os *drivers* fazem com que os sistemas convidados possam acessar o *hardware* da máquina real. Com isso, a comunicação fica muito mais eficiente.

Alguns dos *softwares* de virtualização completa mais populares são o VMware, VirtualBox e [KVM](#).

#### 2.6.4 VM plena

Em uma **VM** plena, ou paravirtualização, não há uma camada de *hardware* que emula os sistemas operacionais convidados, apenas uma interface de aplicativo é entregue para o usuário. Ela consiste em uma técnica que permite que recursos de *hardware* sejam acessados diretamente pelo sistema convidado.

A principal vantagem da utilização da técnica de paravirtualização é que testes realizados pelo hypervisor não são necessários. Pelo fato de ser possível o acesso direto aos recursos de *hardware* pelo sistema convidado, alguns mecanismos de trap de **CPU** são desnecessários, o que torna possível um ganho significativo de desempenho e processamento.

Algumas desvantagens deste tipo de técnica é que o sistema operacional precisa ser modificado, onde um monitor de máquinas virtuais deve ser executado. O monitor de máquinas virtuais Xen é o mais popular utilizado em paravirtualização (**SILVA, 2007**).

#### 2.6.5 *Software-defined as a data center*

O **SDDC** consiste na criação de um data center virtual com tendências na utilização de sistemas distribuídos baseados em arquiteturas web, com auxílio de protocolos simples e comunicação através de *Application Programming Interface* (**API**)s (**SOUZA, 2019**).

#### 2.6.6 *Software-defined Storage*

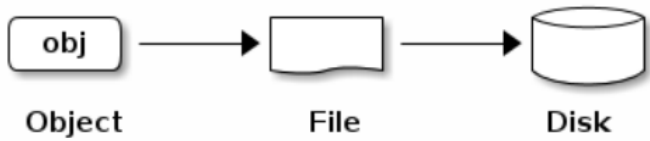
O **SDS**, ou armazenamento definido por software em português, consiste em uma arquitetura de armazenamento de dados distribuída independente do *hardware*. O **SDS** é mais ágil e econômico do que as abordagens tradicionais e possibilita que a capacidade de armazenamento seja baseada em escalabilidade horizontal, ampliando rapidamente de forma mais econômica e flexível (**SOUZA, 2019**).

#### 2.6.7 Ceph

Ceph é uma plataforma de *software* de código aberto utilizada para armazenar grandes quantidades de usuários e de dados. Ele é compatível com os sistemas de arquivos *fourth extended filesystem* (**EXT4**) e *Extended Filesystem* (**XFS**). Nele, os dados são armazenados em forma de objeto, que podem ser replicados através do *Ceph Object Storage Daemon* (**OSD**) para garantir segurança e alta disponibilidade dos dados. Os objetos são armazenados como um arquivo no sistema de arquivos, conforme mostra a figura 3. Através de um algoritmo denominado **CRUSH**, os nós de um *cluster* recebem os dados dinamicamente e periodicamente, para assim reduzir a latência de leitura e escrita nas informações contidas nos blocos de dados. O Ceph utiliza um monitor de *cluster*, que serve para localizar a cópia mais recente dentro do *cluster* (**REIS, 2017**).



Figura 3 – Armazenamento de dados no Ceph.



Fonte: (REIS, 2017)

O [CephFS](#) armazena os dados como objetos. Cada objeto possui seu identificador, seus dados binários e seus metadados. Os metadados consistem em um conjunto de campos contendo nome/valor, que podem ser utilizados para armazenar atributos, por exemplo, dono do arquivo, data de criação, data da última modificação, outros. A figura 4 melhor demonstra os campos de um objeto no [CephFS](#):

Figura 4 – Objeto no [CephFS](#).

ID	Binary Data	Metadata
1234	0101010101010100110101010010 0101100001010100110101010010 0101100001010100110101010010	name1 value1 name2 value2 nameN valueN

Fonte: ([Ceph Documentation](#), 2016)

2.6.8 Rook

Rook é um operador de armazenamento nativo para nuvem utilizado em Kubernetes e que possui código aberto. Sua principal função é automatizar tarefas de um administrador de armazenamento, tais como implantação, inicialização, configuração, provisionamento, dimensionamento, atualização, migração, recuperação de desastres, monitoramento e gerenciamento de recursos.

O Rook usa o poder da plataforma Kubernetes para fornecer seus serviços por meio de um Operador Kubernetes para cada provedor de armazenamento ([AUTHORS](#), 2023).

A principal função do Rook é orquestrar diversas soluções de armazenamento, cada uma delas com seu operador Kubernetes. Com ele, é possível escolher qual o melhor provedor de armazenamento para um cenário em questão. Então, o Rook garante que todos eles funcionem bem no Kubernetes. O principal provedor de armazenamento para o Rook é o Ceph. Outros conhecidos estão em desenvolvimento, entre eles, Cassandra e NFS.



### 2.6.9 Kubernetes

Segundo [The Kubernetes Authors \(2021d\)](#), Kubernetes, também conhecido como k8s, é um sistema de código aberto para automatizar a implantação, o dimensionamento e o gerenciamento de aplicações em contêineres em múltiplos servidores. Além disso, ele elimina grande parte dos processos manuais necessários para implantar e escalar as aplicações em contêineres. Com a utilização de kubernetes, é possível agrupar os *hosts* executados em contêineres linux em *clusters* e gerenciá-los com facilidade e eficiência. Cada *cluster* pode incluir *hosts* em nuvem pública, nuvem privada ou nuvem híbrida, sendo ideal para aplicações em nuvem que exigem escalabilidade.

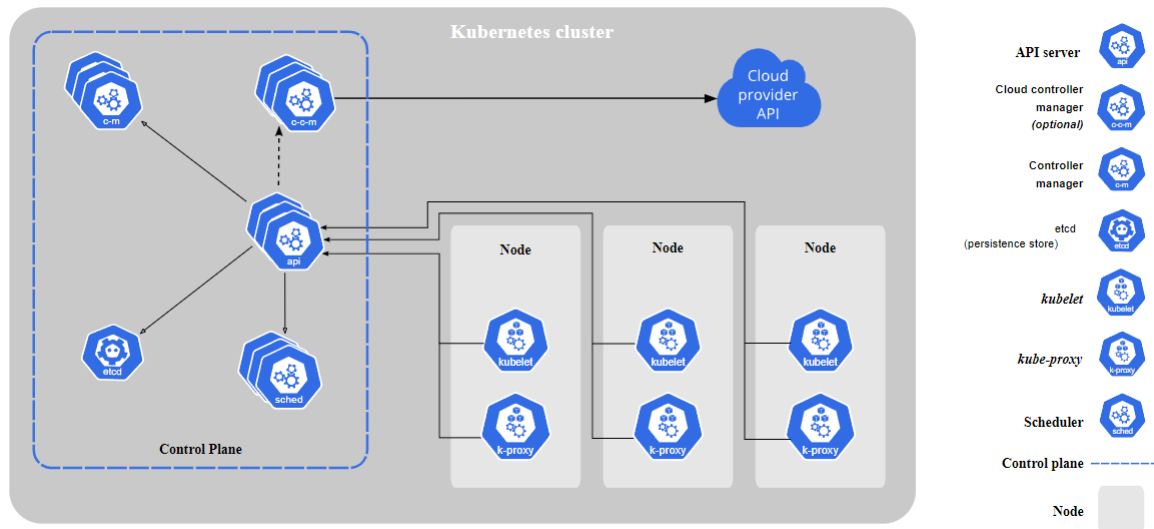
Segundo [Red Hat \(2021\)](#), a utilização de kubernetes possibilita:

- Orquestrar contêineres em vários *hosts*.
- Aproveitar melhor o *hardware* para maximizar os recursos necessários na execução das aplicações corporativas.
- Controlar e automatizar as implantações e atualizações de aplicações.
- Montar e adicionar armazenamento para executar aplicações com monitoração de estado.
- Escalar rapidamente as aplicações em contêineres e recursos relacionados.
- Gerenciar serviços de forma declarativa, garantindo que as aplicações sejam executadas sempre da mesma maneira como foram implantadas.
- Verificar a integridade e autorrecuperação das aplicações com posicionamento, reinício, replicação e escalonamento automáticos.

Um *cluster* kubernetes possui vários componentes, que podem ser divididos em dois tipos, componentes *master* e componentes *node*. A figura 5 mostra um diagrama com estes componentes.

Os nós de trabalho hospedam os Pods. Os pods são as menores unidades de computação implantáveis que você pode criar e gerenciar no Kubernetes [The Kubernetes Authors \(2021a\)](#). Já o *control plane* tem a tarefa de gerenciar os nós de trabalho e os pods em um *cluster*. O *control plane* normalmente é executado em vários computadores; e um *cluster* geralmente executa vários nós. Isso fornece tolerância a falhas e alta disponibilidade.

Conforme podemos visualizar na figura 5, os componentes de um *control plane* são os seguintes:

Figura 5 – Componentes de um *cluster* kubernetes.

Fonte: (The Kubernetes Authors, 2021b)

- **API server:** expõe a **API** do Kubernetes. É a interface para comunicação com o servidor através de uma *Representational State Transfer (REST) API*, para que assim seja possível a comunicação com o plano de controle do Kubernetes.
- **Cloud controller manager** (opcional): incorpora a lógica de controle específica da nuvem. Permite vincular seu *cluster* à **API** do provedor de nuvem. É utilizado para separar os componentes que interagem com essa plataforma de nuvem dos componentes que interagem apenas com seu *cluster*. Assim como o *controller manager*, combina vários *loops* de controle logicamente independentes em um único binário que você executa como um único processo. É possível executar mais de uma cópia para melhorar o desempenho ou ter tolerância a falhas.
- **Controller Manager:** é nele que roda o controlador do processo. Cada controlador é um processo separado e todos eles são compilados em um único binário e executados em um único processo.
- **etcd:** armazenamento de valor de chave consistente e altamente disponível. É utilizado como armazenamento de apoio do Kubernetes para os dados de um *cluster*. Caso o *cluster* Kubernetes utilize etcd como armazenamento de apoio, um plano de *backup* para esses dados é necessário. Vale ressaltar que o nome “etcd” surgiu de uma convenção de nomenclatura da estrutura de diretórios do Linux, onde existe uma pasta chamada “/etc” e o “d” refere-se a “distributed” (IBM, 2023).
- **Scheduler:** observa os Pods recém-criados que não possuem nenhum nó atribuído. Então, seleciona um nó e os executa.

### ***Node components:***

- Kubelet: agente executado em cada node (nó) do *cluster*; garante que os contêineres estejam sendo executados em um Pod; não gerencia contêineres que não foram criados pelo Kubernetes.
- Kube-proxy: é um proxy de rede executado em cada nó do *cluster*. Utiliza a camada de filtragem de pacotes do sistema operacional se ele estiver disponível. Mantém regras de rede em nós e permite a comunicação de rede com seus pods, a partir de sessões de rede dentro ou fora do *cluster*. Do contrário, o kube-proxy encaminha o próprio tráfego.

De acordo com [The Kubernetes Authors \(2021c\)](#), A [API REST](#) é a estrutura fundamental do Kubernetes. Todas as operações e comunicações entre componentes e comandos de usuário externo são chamadas [REST API](#) que o [API Server](#) manipula. Consequentemente, tudo na plataforma Kubernetes é tratado como um objeto de uma [API](#) e tem uma entrada correspondente na [API](#). Portanto, a interação entre os componentes se dá através de comandos padronizados, onde todos os recursos são fornecidos através de [API](#) em HTTP.

## 2.7 Escalabilidade

Escalabilidade consiste em uma medida utilizada para medir o desempenho de um sistema computacional. Normalmente, ela está associada a um aumento na quantidade de *hardware* para melhorar o desempenho de um sistema. É possível adicionar *hardware* em um sistema de duas maneiras e, consequentemente, melhorar a escalabilidade do sistema, que pode ser uma escalabilidade horizontal ou vertical. A escalabilidade horizontal consiste em aumentar a quantidade de máquinas físicas ou virtuais para melhor atender a demanda do sistema. Nela é possível aumentar recursos como memória, *Hard Disk* (HD) e [CPU](#). Ao invés de melhorar os recursos de um determinado servidor, pode-se construir uma nova instância do mesmo e fazer um balanceamento de carga entre estas e outras instâncias que forem criadas para processar uma aplicação. Um pré-requisito deste tipo de aplicação é que a ela seja construída de forma distribuída, onde a mesma pode ser processada por um cliente e não importando qual dos servidores a requisição seja processada. Por outro lado, a escalabilidade vertical aumenta a quantidade de recursos físicos ([CPU](#), memória, armazenamento) de uma única máquina, para que assim possa melhorar a capacidade da mesma. Recursos como memória, [HD](#) e [CPU](#) são aumentados dentro da mesma máquina física ou virtual.

[RAVANÊDA \(2014\)](#) diz que é possível definir o tipo de escalabilidade que se deseja executar, de acordo com o nível de automação que se deseja aumentar na hora de alocar um determinado recurso. Sendo assim, outros quatro tipos de escalabilidade podem ser adotados, escalabilidade forçada, manual, configurada ou automática.

A escalabilidade forçada ocorre quando uma organização decide aumentar a quantidade de recursos, normalmente servidores ou equipamentos de rede, por um determinado período. Esse aumento ocorre em virtude de previsão de aumento de demanda em determinado período. Porém, essa previsão pode não ser muito precisa, onde dificilmente a demanda real será suprida nos exatos períodos de tempo em que será requisitado aumentar os recursos.

Têm-se uma escalabilidade manual quando um operador observa manualmente um aumento de demanda. Então, o mesmo efetua a alocação de novos recursos baseado em sua experiência, na leitura de dados de seu sistema. Isso faz com que o aumento na quantidade de recursos seja mais eficiente que na escalabilidade forçada, pois somente será alocado recurso quando realmente ele for necessário. Quando não houver mais demanda, ele será removido.

Já na escalabilidade configurada, *scripts* baseados em leitura de estatísticas e métricas do sistema são utilizados para automatizar as alocações e desalocações de recursos. Por exemplo, se a demanda ultrapassar 90%, novos recursos devem ser alocados. E se baixar de 30%, eles devem ser desalocados. Isso automatiza as tarefas de alocação e desalocação de recursos que o operador tinha que fazer manualmente na escalabilidade manual.

E por último, a escalabilidade automática onde a demanda ofertada é exatamente igual a demanda real. Nesse cenário, não existe desperdício de recursos. Mesmo com uma escalabilidade configurada, há um pouco de desperdício de recursos, pois ele não é exatamente aquilo que se necessita em relação a demanda real, por melhor que seja configurados os *scripts*. Porém, na escalabilidade automática, formas inteligentes de alocação e desalocação de recursos são empregados. O sistema utiliza simulações e leituras locais para decidir o momento exato de realizar as mudanças.

## 2.8 Pagamento pontual (*PAY-AS-YOU-GO*)

O pagamento pontual consiste em pagar pontualmente por uma unidade de recurso que se deseja utilizar, de forma a gastar a menor quantidade possível desse recurso.

Por exemplo, ao invés de pagar por um plano 100 GB em um mês, utiliza-se um plano de 2 MB por hora. Com isso, é possível utilizar-se apenas dos recursos realmente necessários. Imagina se nesse cenário fosse preciso apenas utilizar armazenamento de 2 MB por hora durante apenas 4 dias no mês. A quantidade de recursos necessários seria muito inferior do que no plano de 100 GB por mês, trazendo grande economia.

Esse modelo de “pagamento por uso” é oferecido por grandes organizações, como Google, Azure e Amazon. E o modelo de computação em nuvem que utiliza essa forma de

pagamento é a nuvem pública.

## 3 Princípios da computação em nuvem

Neste capítulo serão apresentadas algumas das principais tecnologias oferecidas quando nos referimos de computação em nuvem. Estas tecnologias são algumas das principais soluções encontradas para resolver problemas envolvendo computação em nuvem e sua utilização como solução para fornecimento de conteúdo pela internet. Algumas destas soluções são o *High Performance Computing* (HPC), *Content Delivery Network* (CDN) e *Peer-to-Peer* (P2P). Outros tópicos abordando a maneira como empresas vendem estas soluções também são abordadas neste capítulo.

### 3.1 Computação como utilidade

Segundo SOUSA (2014b), computação como utilidade é um conceito que diz que computação em nuvem passará a ser visto como um serviço, e não mais como um produto. A necessidade do usuário deixará de ser na aquisição da totalidade de uma determinada solução, passando a pagar apenas por aquilo que ele realmente utiliza, da mesma maneira como ocorre no consumo de água, luz, telefone, etc.

Cada vez mais a ligação entre componentes de *hardware* e aplicações diminuiu. Isso possibilitou que novas formas de descrever sistemas fossem feitas pelos profissionais da área de tecnologia da informação. Componentes de *hardware* que em outros tempos eram incompatíveis agora podem ser unidos. Com isso, uma solução mais eficiente pode ser desenvolvida. Tudo isso ocorreu graças ao desenvolvimento do conceito de SaaS, visto na sessão 2.4.2.

### 3.2 *High Performance Computing* (HPC)

O HPC é um termo que consiste na utilização de supercomputadores ou *clusters* para solucionar problemas mais complexos. Como benefícios, traz um menor tempo de processamento, menor tempo de resposta (latência) e maior largura de banda para a aplicação em que se utilize esta modalidade. Sua utilização deve ser empregada em aplicações de computação intensiva e insensível a dados. De maneira simplificada, é possível explicar o funcionamento de um HPC como um grupo de computadores trabalhando juntos, em grupo (*cluster*), para realizar um trabalho de alta complexidade em um curto período de tempo.

Um HPC pode ser executado tanto localmente, em nuvem ou em modelo híbrido. Por exemplo, uma empresa pode construir uma infraestrutura com um *cluster* de servidores.

Porém, algumas empresas administram e gerenciam esse tipo de serviço, sendo utilizado para isso um modelo *Pay-as-you-go* (2.8).

Um ponto importante a ser mencionado sobre HPC é que algumas empresas vendem *hardwares* dedicados para utilização em aplicações voltados para aplicações com grandes volumes de dados. Existem também alguns tipos de *hardware* dedicado para melhorar o desempenho de sistemas que utilizem HPC, são eles:

- *Graphics Processing Unit* (GPU)s ou Unidade de Processamento Gráfico - Algumas empresas como Devices (2022) e Nvidia (2022) possuem *hardwares* gráficos dedicados para otimizar aplicações com HPC.
- *Field Programmable Gate Array* (FPGA)s ou “Arranjo de Portas Programáveis em Campo” - A Altera Corporation (2007) e a Xilinx Advanced Micro Devices (2022) são dois grandes fabricantes de dispositivos FPGAs, que são *hardwares* dedicados exclusivamente para HPC. Este primeiro ainda traz gráficos sobre o ganho de desempenho dos FPGAs.
- *Tensor Processing Unit* (TPU)s - É uma unidade de processamento proprietário do Google feito para projetos com aprendizagem de máquinas (*machine learning*) e redes neurais. Normalmente é utilizado em conjunto com GPUs e CPUs para melhorar o desempenho de HPCs.

A instalação, configuração e manutenção deste tipo de *hardware* é complexa e requer mão-de-obra qualificada. Por este motivo, é comum que grandes empresas ofereçam HPC como serviço de nuvem pública. Clientes apenas contratam o serviço e deixam que o fornecedor se preocupe com o tipo de *hardware* e como ele deve ser entregue para o cliente. Amazon (2022b), Microsoft (2022b) e Oracle (2022) são exemplos de grandes empresas que oferecem esse tipo de serviço em nuvem.

### 3.3 Arquitetura Orientada a Serviços (AOS)

A Arquitetura Orientada a Serviço (AOS), ou *Service-Oriented Architecture* (SOA), é uma arquitetura em que os componentes são implementados como serviço, são flexíveis, comunicam-se por mensagens e podem ser orquestrados de maneira amarrada, podendo ser disponibilizados para usuários na nuvem como um serviço.

A lógica de uma AOS é a sua utilização para a automação de tarefas. Por esse motivo, é necessário que seus componentes sejam modulares e que possam ser facilmente acoplados. O objetivo disso é ter uma arquitetura de *software* melhor organizada e possibilitar, segundo SOUZA (2011), os seguintes benefícios:

- Melhor integração - inter operabilidade entre o conjunto de soluções utilizadas;
- Reuso nativo - possibilidade de reaproveitar as soluções já em uso;
- Arquiteturas e soluções racionalizadas - a solução possui um ambiente otimizado, onde somente o que realmente é necessário passa a fazer parte da arquitetura;
- Aproveitamento do investimento legado - custo e esforço para integrar a solução legada relativamente baixos;
- representação de dados padronizada em *Extensible Markup Language* (XML) - com XML, a representação dos dados fica padronizada, diminuindo assim esforços para análise, desenvolvimento e documentação de um projeto;
- Investimento focado em infra estrutura de comunicação - graças a inter operabilidade, o custo de escalonamento da infra estrutura é reduzido;
- Melhores alternativas de desenvolvimento - também graças a inter operabilidade, é possível escolher a melhor alternativa em determinado contexto;
- Agilidade Organizacional.

Segundo RODRIGUES (2011), nesse tipo de arquitetura, as interfaces desse tipo de arquitetura são definidas em um formato XML, os serviços são definidos em *Web Services Description Language* (WSDL) e também suas aplicações acessam os serviços em um *Universal Description, Discovery and Integration* (UDDI).

### 3.4 Banco de dados

Normalmente, espaços de armazenamento em nuvem são vendidos na forma de PaaS. Então, dessa forma, para construir uma aplicação, é necessário contratar também um serviço de banco de dados, que pode ser *Structured Query Language* (SQL) ou *Not only Structured Query Language* (NoSQL), que podem ser associados a dois tipos de banco de dados (BD), relacional e não relacional, respectivamente.

Segundo RAMAKRISHNAN e GEHRKE (2011), é importante que um Sistema de Gerenciamento de Bases de Dados (SGBD) apresente quatro características importantes para que os dados armazenados no banco sejam apresentados sem erros de sistema. Essas propriedades são comumente chamadas de Propriedades Atomicidade, Consistência, Isolamento e Durabilidade (ACID), e são elas:

1. Atomicidade - Cada usuário deve ser capaz de realizar execuções atômicas. Isso consiste em executar a todas as suas transações ou nenhuma delas.



2. Consistência - Os usuários devem garantir a consistência dos dados em todas as transações.
3. Isolamento - Uma transação não pode interferir em outra transação em execução em um banco de dados. Ou seja, as operações de transação devem ser isoladas umas das outras.
4. Durabilidade - O [SGBD](#) deve informar ao usuário que sua transação foi efetuada com sucesso. Então, os efeitos dessa transação devem persistir mesmo que o sistema falhe antes que as alterações sejam gravadas em disco.

### 3.4.1 Banco de dados relacional

No banco de dados relacional, a forma como os dados são definidos, armazenados, manipulados e recuperados é estruturada, através de tabelas, com linhas e colunas. A maneira como duas ou mais tabelas se relacionam é bem definida nesse tipo de banco de dados. O acrônimo [SQL](#) está relacionado a este tipo de base de dados.

### 3.4.2 Banco de dados não relacional

Já no banco de dados não relacional, o conjunto de dados não segue um padrão como os relacionais. Esse tipo de banco de dados serve para armazenar grandes quantidades de dados e não é necessária uma padronização destes dados. O acrônimo [NoSQL](#) está relacionado a este tipo de base de dados.

Segundo [TENORIO \(2022\)](#), *Remote Dictionary Server* ([REDIS](#)) é um código de fonte aberto que pode ser utilizado como banco de dados, *cache* e *message broker*. Importante destacar também que o [REDIS](#) é muito utilizado como ferramenta de cache por empresas como Github, Twitter e Instagram.

Com o [REDIS](#), é possível sua utilização como servidor para estruturas de dados, como *strings*, dicionários, listas, e informações geoespaciais. Ele foi desenvolvido em linguagem de programação C e oferece suporte apenas para plataformas Linux e Mac OS. Também é possível sua utilização em ambiente Windows (a partir de sua versão 10) com o recurso *Windows Subsystem For Linux* ([WSL](#)). Com ele, é possível utilizar comandos do Linux dentro do próprio Windows.

Bancos de dados não relacionais são do tipo chave-valor e que tem como vantagens o fato de serem uma ferramenta *opensource* amplamente difundida na comunidade de desenvolvedores, possuem facilidade de operação e também apresentam a possibilidade de configurar tempo de vida (TTL do inglês Time to Live).

### 3.4.3 Persistência de dados

No campo de ciências da computação, persistência de informação consiste na “sobrevivência” de um determinado estado ao processo a que ele foi criado. Normalmente, essa característica se refere em “salvar” algo que esteja sendo utilizado pela memória [RAM](#) em alguma unidade de armazenamento não volátil, como discos rígidos ou memória *flash*.

Editores de imagem e processadores de texto atingem a persistência salvando o estado em arquivos. Fato é que as informações utilizadas precisam ser armazenadas de maneira confiável. A essa característica dá-se o nome de persistência. Há diversas formas de se enquadrar um sistema como sendo persistente. Alguns exemplos de persistência que podemos citar, entre eles as memórias persistentes (*Persistent Memory* ([PMem](#))), em que a informação fica salva mesmo após uma queda de energia. Outro exemplo é a memória *cache* também é possível persistir informações. Nela, informações recentes e frequentemente utilizadas são armazenadas e utilizadas pelo processador para se obter um acesso mais rápido. Além destas, existem outras diversas técnicas de persistência utilizadas para armazenar informações.

E por último, para exemplificar, temos os sistemas *web* também necessitam que seus dados sejam persistidos no banco de dados. É necessário que seja possível uma restauração de dados sem perda de informação, caso algum problema ocorra no momento de salvar dados no sistema de banco de dados. Em outras palavras, o armazenamento de dados em um sistema de banco de dados deve ser estável. Isso é feito através de um conjunto de *softwares* capazes de gerenciar as interações do usuário em um banco de dados. Esse conjunto de *softwares* dá-se o nome de [SGBD](#), que utiliza técnicas de persistência de dados em um banco, utilizando para isso, além de algumas técnicas citadas no capítulo anterior, mas também outras. Um [SGBD](#) utiliza também técnicas de persistência em nível de *softwares*, que serão descritas abaixo.

Através de alguns mecanismos de manipulação de informação, um [SGBD](#) consegue persistir as informações no banco, de forma segura e estável. O acrônimo *Create, Read, Update and Delete* ([CRUD](#)) resume bem um conjunto de operações de manipulação em um banco de dados relacional. As operações de [CRUD](#) em aplicações de bases de dados relacionais podem ser mapeadas da seguinte maneira:

- *CREATE* - comando *INSERT*. Cria ou adiciona novos registros em uma tabela;
- *READ* - comando *SELECT*. Comando utilizado para ler um ou mais registros em uma tabela;
- *UPDADE* - comando *UPDATE*. Atualiza um ou mais registros em uma tabela.
- *DELETE* - comando *DELETE*. Remove um ou mais registros em uma tabela.

Contudo, cada banco de dados deve ser compartilhado e gerenciado em diferentes plataformas. Isso criou alguns problemas para realizar operações em tabelas de dados que utilizem plataformas distintas. Para isso, [CLARO \(2017\)](#) traz uma solução para o problema. Em sua tese, trazendo o conceito de "Persistência Poliglota". Esse termo significa que podemos aproveitar o melhor de cada linguagem ou ferramenta, e ainda, adequar a interação entre elas. Ou seja, o relacionamento entre diferentes bases de dados, de diferentes aplicações, em diferentes formatos ou linguagens é possível, através da "Persistência Poliglota".

Para isso, o autor acima citado aborda o conceito de [REST Services](#) e [REST API](#). [REST](#) é um acrônimo para, em português, Representação de Estado de Transferencia. Uma [API](#) ou um serviço baseados em [REST](#) possui uma arquitetura orientada para sistemas distribuídos. Essa arquitetura utiliza o protocolo *Hypertext Transfer Protocol* ([HTTP](#)) para realizar a requisição, inserção e edição de dados nos bancos. E isso também é feito através de operações de CRUD, conforme segue:

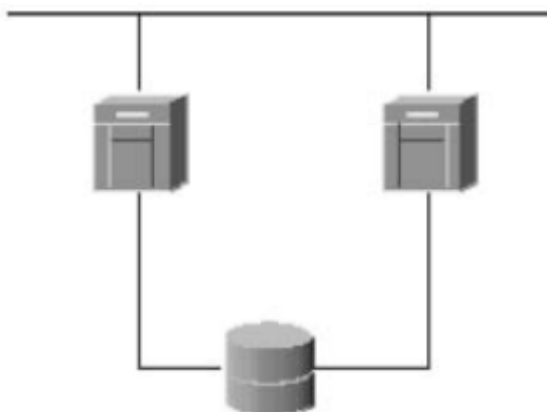
- *CREATE* - comando *HTTP PUT*. Cria ou adiciona novos registros em uma tabela;
- *READ* - comando *HTTP GET*. Comando utilizado para ler um ou mais registros em uma tabela;
- *UPDADE* - comando *HTTP POST*. Atualiza um ou mais registros em uma tabela.
- *DELETE* - comando *HTTP DELETE*. Remove um ou mais registros em uma tabela.

Em resumo, com a utilização de uma [API REST](#), é possível uma escrita em NodeJS, além de ser possível montar e manipular dados de diversas tabelas não padronizadas entre si. NodeJS é um *software* de código aberto, utilizado para permitir a utilização de códigos escritos na linguagem de programação *JavaScript* fora de um navegador *web*.

Outro ponto importante a ser considerado referente a persistência de dados é quanto aos tipos de armazenamento de informação em disco e a maneira como os dados são compartilhados. Uma maneira mais simples de se classificar os tipos de persistência é com base no tipo de memória utilizada pelos equipamentos e a maneira com que as informações são persistidas nas mesmas. Sendo assim, a persistência pode ocorrer na memória principal, na memória cache ou nas memórias secundárias.

A persistência na memória principal ocorre através de interações entre processador e memória [RAM](#). Basicamente, uma memória principal é o dispositivo onde o “programa principal” será executado e onde suas informações são armazenadas, para que o processador vá buscando, instrução por instrução. O processamento destas instruções é feito pelo processador, com auxílio da memória [RAM](#), que armazena temporariamente os dados

Figura 6 – Armazenamento com compartilhamento total de disco.



Fonte: (RAMOS, 2009)

referente ao processamento destas instruções. Contudo, este tipo de memória não é a mais indicada para gravar permanentemente (persistir) informações relevantes.

Caso a persistência ocorra na memória *cache*, uma quantidade pequena de memória que fica próxima ao processador. Ela serve de um acesso intermediário entre processador e memória RAM. A memória RAM normalmente tem uma capacidade muito maior que a memória *cache*, e suas funções são semelhantes. Porém, na memória *cache* normalmente se armazenam dados referentes a processos frequentemente utilizados pelo processador. Desta forma, seu acesso fica mais rápido. Porém, este tipo de memória possui pouca capacidade de armazenamento, até mesmo se for comparar com a memória RAM.

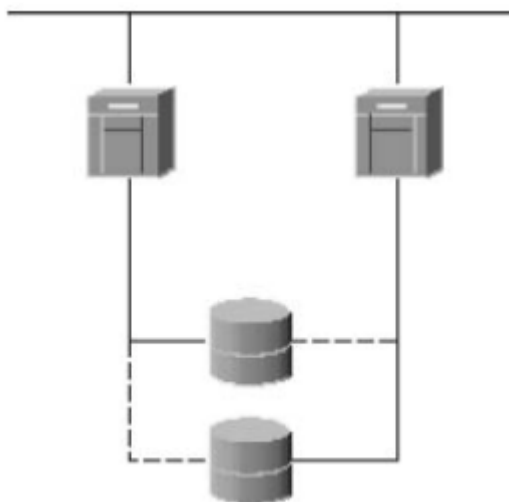
Porém, a persistência pode ocorrer também na memória secundária, que serve para armazenar permanentemente grandes volumes de informação. Esse tipo de memória é muito mais barata que as anteriores, e a tecnologia envolvida na fabricação dos equipamentos é através de meios magnéticos, *Electrically-Erasable Programmable Read-Only Memory* (EEPROM), FLASH, *Solid-State Drive* (SSD) e fitas magnéticas. Pelo seu baixo custo, é comum a utilização de dispositivos com grandes volumes de armazenamento deste tipo.

Uma outra maneira de classificar tipos diferentes de armazenamento (ou persistência) de dados é apresentada por RAMOS (2009), conforme o tipo de compartilhamento dos dados entre servidores. Ele pode ser Compartilhamento Total, Sem Compartilhamento e com Espelhamento.

No compartilhamento total, todos os servidores utilizam o mesmo disco ou unidade de armazenamento. Problemas como corrupção de dados ocorrem quando mais de um servidor está ativo nesse tipo de armazenamento. A figura 6 mostra 2 (dois) servidores compartilhando o mesmo disco.

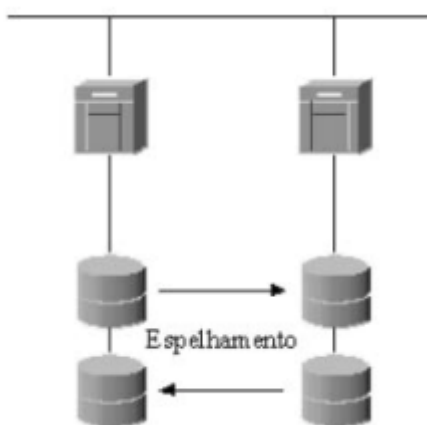
Quando não há compartilhamento de informações entre servidores (sem compartilhamento), cada servidor possui sua própria unidade de armazenamento. Nesse caso,

Figura 7 – Armazenamento sem compartilhamento de disco.



Fonte: (RAMOS, 2009).

Figura 8 – Armazenamento com espelhamento de disco.



Fonte: RAMOS (2009).

quando um dos servidores vier a apresentar falha, o servidor “reserva” acessará a unidade de armazenamento do nó que apresentou defeito. A figura 7 mostra 2 (dois) servidores, cada um com sua própria unidade de armazenamento. Em caso de falha, um servidor poderá acessar a unidade de armazenamento de outro que venha a apresentar falha.

Caso exista compartilhamento através de espelhamento de discos, não há compartilhamento de discos entre nós representados por cada conjunto de servidor e unidades de armazenamento. Cada servidor possui sua própria unidade de armazenamento e suas réplicas, conforme podemos ver na figura 8.

Quando um servidor possui mais de uma unidade de armazenamento, podemos dizer que há um *array* de discos. Existem formas diferentes de realizar o espelhamento de discos, visando melhorar o desempenho e a confiabilidade do sistema. Essas formas

Figura 9 – Comparação entre RAIDs.

Nível de RAID	Disponibilidade do dado	Desempenho de leitura	Desempenho de gravação	Desempenho na reconstrução dos dados	Quantidade mínima de discos	Recomendação de uso
0	Baixa	Muito bom	Muito bom	-	N	Dados não críticos
1	Excelente	Muito bom	Bom	Bom	2 (N=1)	Pequenas bases de dados, logs, dados críticos
5	Boa	Leitura sequencial: Boa/Leitura transacional: Muito boa	Razoável	Ruim	N+1 (N≥2)	Bases de dados e aplicações com alta intensidade de leitura

N = Número de discos no array

Fonte: RAMOS (2009).

distintas de organizar os discos em espelhamento são popularmente conhecidos como RAID (*Redundant Array of Inexpensive Drives*). Segundo RAMOS (2009), as diferentes formas de RAID são RAID 0, RAID 1, RAID 2, RAID 3, RAID 4, RAID 5, RAID 6, RAID 0 + 1, RAID 1 + 0, RAID 50 e RAID 100.

RAID 0, RAID 1 e RAID 5 estão relacionados a *clusters*, além de ser os mais populares em meio acadêmico. A figura 9 mostra as diferenças entre estes 3 (três) tipos de RAID.

Conforme foi visto, há diversos níveis de persistência de informações em sistemas. É importante ressaltar que estas tecnologias possuem um tempo de vida útil, de forma que necessitam ser substituídos quando a tecnologia expira ou quando apresentam defeitos. Essas substituições geram custos que devem ser levados em consideração quando se projeta um banco de dados, ou um sistema como um todo.

Atualmente, há grandes empresas do ramo de computação em nuvem que vendem sistemas de armazenamento de dados persistentes e confiáveis, em formato de serviço. Quando estes serviços são contratados, o usuário não tem a necessidade de se preocupar com a persistência de dados. Eles são de responsabilidade da empresa contratada. A única coisa que os usuários precisam se preocupar é com a forma de utilização destes serviços e com a interação entre um usuário e o sistema contratado. Normalmente isso ocorre através de APIs, com comandos simples de se utilizar, como por exemplo, aquelas citadas no item 3.4.3. Com a popularização deste tipo de serviço na atualidade, o custo deste tipo de serviço vem se tornando cada vez menor e vale a pena ao menos a verificação dos custos deste tipo de serviço, pelo tempo e condições minimamente necessárias para o uso que se deseja contratar. Alguns exemplos mais populares deste tipo de serviço são o *Simple Storage Service (S3)* da Amazon (Amazon (2022a)), o *Azure Blob Storage* da Microsoft (Microsoft (2022a)) e o *Cloud Storage* da Google (Google (2022)).

### 3.5 *Content Delivery Network* (CDN)

O **CDN** consiste no agrupamento de *data centers* espalhados pelo mundo. Um determinado conteúdo apenas é replicado para um destes *data centers* quando necessário. E para tornar o sistema mais eficiente, normalmente se replica um conteúdo para um *data center* que esteja geograficamente mais próximo de vários usuários, para assim diminuir o tempo de latência na hora de baixar o conteúdo. Quando o conteúdo é fornecido para o usuário por uma réplica, isso significa que a demanda pelo servidor deixou de acontecer, fazendo com que servidores com maior capacidade não sejam necessário, trazendo redução de custos com infraestrutura física.

### 3.6 *Peer to peer* (P2P)

O **P2P** trata-se de uma rede virtual de máquinas interligadas, onde usuários compartilham arquivos ou partes dele uns com os outros, sem a necessidade de gastar grandes quantidades de recursos armazenando dados. Os próprios usuários armazenam arquivos e os disponibilizam para outros usuários, reduzindo a necessidade de gastos maiores com servidores e tornando o sistema como um todo muito mais flexível e expansível. Em uma arquitetura cliente-servidor, a comunicação ocorre entre um cliente e o servidor. Quando o número de usuários aumenta, aumenta a demanda pelo servidor. Porém, no **P2P**, a comunicação ocorre da forma cliente-cliente. Na verdade, um cliente que dispõe de um conteúdo solicitado por outro cliente fará a função de servidor. Desta forma, a rede pode funcionar sem a necessidade de um servidor de conteúdo para os clientes. Os próprios clientes que possuem um determinado conteúdo atuam como agente de transferência do mesmo para outro cliente que o requisiute.

### 3.7 CDN-P2P

Trata-se de uma solução que une as tecnologias **CDN** e **P2P**. Com ela, o conteúdo fica armazenado em uma rede **CDN**. Porém, quando houver aumento em uma demanda por conteúdo em uma determinada região, os próprios usuários que já possuem o arquivo passam a fornecê-lo a outros usuários. Dessa forma, é possível pré disponibilizar um determinado conteúdo de maneira segura e caso a demanda aumente, não é necessário gastar mais recursos com novos servidores. Aqui os próprios usuários que já obtiveram o conteúdo do servidor passam agora a atuarem como se fossem um novo servidor e fornecem o mesmo a outros usuários que ainda não o obtiveram.

## 4 Contratação de serviço de nuvem em órgãos públicos

Conforme visto no capítulo anterior, implantar um sistema de computação em nuvem requer enorme preocupação com diversos tópicos complexos. Um sistema em nuvem pode ser simples ou complexo, que vai desde uma pequena página web a um bloco de ferramentas e serviços na internet, todos interligados entre si.

É comum que gestores ou pessoas com cargos elevados, normalmente responsáveis por tomadas de decisão, não tenham conhecimento aprofundado em computação em nuvem. O que este trabalho se propõe é servir de suporte nestes casos, auxiliando na tomada de decisão com o caminho a ser seguido caso algum sistema em nuvem seja necessário como solução de uma demanda tecnológica.

### 4.1 Legislação

A preocupação com legislações e instruções normativas é algo relevante em qualquer atividade profissional. Especialmente quando nos referimos a Instituições Públicas do Governo Federal que fornecem serviços a uma grande quantidade de pessoas. Por isso, esta seção tem como objetivo alertar sobre algumas normas que devem ser respeitadas na hora de fornecer ou contratar serviços em nuvem em entidades da esfera Federal.

Segundo [SOUZA \(2019\)](#), após a idealização do programa Governo Eletrônico - eGOV (substituído em 2016 pela *Estratégia de Governança Digital (EGD)*), uma série de iniciativas surgiu com o objetivo de melhorar a atuação da [TIC](#) do Governo Brasileiro. Para auxiliar na busca por documentos relevantes e relacionados com computação em nuvem em Instituições Públicas na esfera Federal (não restringindo a participação, por adesão voluntária, de outros Poderes e esferas de governo), segue abaixo um breve resumo e documentação de referência para os mesmos:

1. [EGD](#) - A Estratégia de Governo Digital é um documento atualizado a cada triênio que expõe objetivamente as metas para a transformação digital de serviços no âmbito da administração pública federal (direta, autárquica e fundacional). É uma forma de os governos oferecerem transparência à sociedade em geral e aos órgãos de fiscalização e controle quanto às ações desenvolvidas ([Ministério da Gestão e da Inovação em Serviços, 2022](#)).
2. ePING - Outro documento importante que regulamenta a utilização da Tecnologia de Informação e Comunicação no Poder Executivo Federal é o **e-PING! (e-PING!)**



(Ministério do Planejamento, Desenvolvimento e Gestão, 2018). Este documento, em sua versão de 2018, assim diz:

“[...] define um conjunto mínimo de premissas, políticas e especificações técnicas que regulamentam a utilização da **TIC** na interoperabilidade de serviços de governo eletrônico, estabelecendo as condições de interação com os demais Poderes e esferas de governo e com a sociedade em geral.”

3. Guia Livre: Referência de Migração para Software Livre do Governo Federal - visa ser uma referência para processos de Migração para Software Livre no Governo Federal, bem como em qualquer outro nível de governo ou esfera de poder, que porventura necessitem utilizar tal material como referencial ou desejem planejar e executar seus processos de migração com base de sustentação em casos concretos de estratégias já implementadas (Ministério do Planejamento, Orçamento e Gestão, 2022).
4. Boas práticas, orientações e vedações para contratação de Serviços de Computação em Nuvem - é *“um documento de Boas práticas, Orientações e Vedações tem força normativa legal, estando vinculado à Portaria MP/STI nº 20”*, do ano de 2016. Ele traz algumas orientações para Órgãos Públicos visando *“redução de custos, elasticidade, redução da ociosidade dos recursos, agilidade na implantação de novos serviços, foco nas atividades finalísticas do negócio e uso mais inteligente da equipe de TI”* (Ministério do planejamento, orçamento e gestão, 2022).

#### 4.1.1 Documentos norteadores do IFSC

O **IFSC** também possui algumas leis que devem ser levadas em consideração quando se deseja contratar produtos e serviços, em especial, quando este envolve computação em nuvem, tema deste trabalho. Nesse sentido, será apresentado abaixo dois documentos elaborados por gestores de diversas áreas no **IFSC**. Uma delas é de extrema importância, levando em consideração o objetivo deste trabalho, o **PDTIC** 2021/2022.

O **PDTIC** dos anos de 2021/2022, é o instrumento de diagnóstico e gestão dos recursos de **TIC Instituto Federal de Santa Catarina – Reitoria** (2021). Este documento aponta o que está sendo feito atualmente e o que se pretende fazer para melhorar a infraestrutura de tecnologia da informação. Como o temática deste trabalho envolve, principalmente, computação em nuvem, alguns tópicos relevantes presentes no **PDTIC** 2022/2022 serão abordados.

Na página 45 do **Instituto Federal de Santa Catarina – Reitoria** (2021), em seu item 5, mostra algumas ações para manter e expandir a infraestrutura de **TIC** do **IFSC**. Alguns pontos importantes deste item:

- Consta como objetivos deste item a “contratação em nuvem ou criação de nuvem privada para suporte aos câmpus do [IFSC](#)”.
- A justificativa do item informa que os equipamentos do câmpus estão obsoletos e que um plano de capacidade de negócio precisa ser elaborado.
- Consta como indicadores e metas na área para 2021 e 2022, entre elas, a aquisição de 93 switches no valor de R\$ 467.798,35 e a aquisição de 21 servidores totalizando R\$ 478.101,00.
- Outra meta é adotar infraestrutura de nuvem como parte da estrutura tecnológica existente, sendo que o gasto previsto para 2022 é R\$ 43.003,00 para retenção e restauração de backup em nuvem.
- Outro ponto importante a ser explorado como meta é a adesão a alguns itens de serviço em nuvem da Ata de Registro de Preços 11/2021, do Ministério da Economia, sendo que R\$ 73.618,38 já estão em execução em 2022.

Conforme podemos observar anteriormente, os relatórios do [PDTIC](#) 2021/2022 apontam para uma necessidade de aquisição de novos equipamentos para manter serviços em nuvem. Também aponta para uma saturação deste modelo, de modo que um novo plano de capacidade de negócio precise ser elaborado. E como o Ministério da Economia realizou uma contratação unificada para vários Órgãos Públicos da esfera Federal, entre eles o [IFSC](#) e outros Institutos Federais, talvez o modelo de negócio presente na ATA DE REGISTRO DE PREÇOS Nº 11/2021, do Ministério da Economia, aponte para um norte na tomada de decisão no para serviços em nuvem na referida área

#### 4.1.2 Serviços disponíveis na [ARP](#) 11/2021

A [ARP](#) 11/2021 do Ministério da Economia tem o seguinte objeto:

Registro de Preços para a eventual contratação de serviços de tecnologia da informação e comunicação de empresa especializada para prestação de serviços gerenciados de computação em nuvem, sob o modelo de cloud broker (integrador) de multi-nuvem, que inclui a concepção, projeto, provisionamento, configuração, migração, suporte, manutenção e gestão de topologias de serviços em dois ou mais provedores de nuvem pública.

Segundo artigo publicado em [SoftwareOne Brasil](#) (2020), um *cloud broker* (corretor de nuvem) é uma pessoa ou empresa que atua como um corretor de serviços em nuvem, de maneira análoga a um corretor de imóveis. Por exemplo, o corretor de imóveis colhe a necessidade do cliente e então procura a solução que melhor atende a necessidade de seu cliente, pelo menor custo. Um *cloud broker* tem função similar. Porém, ao invés de

Figura 10 – Tabela de preços, especificações e quantitativos.

ITEM	DESCRIÇÃO/ ESPECIFICAÇÃO	Unidade de Medida	Quantidade	Valor Unitário (R\$)
1	Serviços de Computação em nuvem – Infraestrutura como Serviço (IaaS)	Unidade de Serviço de Computação em Nuvem -USN	22.139.832	1,88
2	Plataforma como Serviço - PaaS	Unidade de Serviço de Computação em Nuvem -USN	13.605.864	1,51
3	Software como Serviço - SaaS	Unidade de Serviço de Computação em Nuvem -USN	784.392	0,62
4	Serviço de Gerenciamento e Operação de recursos em nuvem	Unidade - Instância gerenciada por mês	25.636	86,70
5	Serviço de Migração de Recursos Computacionais	Unidade - Instância de Computação migrada	1.404	259,00
6	Serviço de Migração de Banco de dados	Unidade - Instância de Banco de Dados migrada	616	255,00
7	Treinamento	Unidade - Turma de treinamento	68	8.000,00

Fonte: [Ministério da Economia \(2021\)](#)

selecionar os melhores imóveis, o corretor irá fornecer os melhores serviços de nuvem para seu cliente de acordo com a sua necessidade, de maneira eficiente.

Conforme diz o item 7.7, do Estudo Técnico Preliminar (ETP) para o PREGÃO ELETRÔNICO Nº18/2020, do Ministério da Economia, presente em [Ministério da Economia \(2021\)](#), o qual deu origem a [ARP 11/2021](#):

“[...] a utilização dos *brokers* mostra-se como uma opção relevante em se tratando de contratação de serviços de nuvem para o governo federal, permitindo implantar um modelo que minimize os riscos inerentes a referida contratação, considerando que os órgãos da administração pública federal possuem baixa maturidade na contratação e uso de serviços de computação em nuvem, conforme aponta o Acórdão 1.739/2015-P, do Plenário do Tribunal de Contas da União.”

Por esse motivo, o serviço de contratação do tipo *broker* é utilizado pelo Governo Federal.

Portanto, o broker de nuvem contratado por esta [ARP](#) deve indicar para o contratante qual tipo de serviço destes melhor se enquadra ao cenário do contratante.

Os serviços presentes na [ARP 11/2021](#), com seus respectivos valores são os seguintes:

Observação importante: Qualquer órgão ou entidade da administração pública (federal, estadual ou municipal) pode realizar a adesão a [ARP 11/2021](#).

## 4.2 Processo de contratação pública

Para realizar uma compra ou contratação de serviços em Órgãos Públicos, a regra é a licitação, devendo esta obedecer a uma série de critérios legais e formais para definir

qual empresa fornecerá a solução para a entidade contratante. Neste trabalho, será citado duas formas de contratação através de licitação, aquela do tipo técnica e preço e outra do tipo menor preço. Porém, em alguns casos, é possível dispensar alguns procedimentos formais para uma contratação.

Na licitação do tipo técnica e preço, não somente o preço será o critério de escolha do fornecedor, mas também os critérios técnicos que serão levados em conta na hora de escolher uma proposta. Um sistema de pontuação deve ser muito bem documentado para que os fornecedores possam expor as ideias para que uma comissão possa julgar e atribuir uma nota para sua solução. A pontuação final, neste caso, será um combinado entre melhor técnica e menor preço. Tudo muito bem documentado. Normalmente, licitações do tipo técnica e preço são realizadas na modalidade Concorrência Pública.

Em licitação do tipo menor preço, são definidas criteriosamente quais especificações o serviço deve ter, como no caso acima, mas agora a forma de escolha do fornecedor será através do menor preço apenas. Se as especificações do serviço puderem ser muito bem detalhadas, então esta forma pode ser utilizada. Normalmente, para as licitações do tipo menor preço, recomenda-se a escolha da modalidade de licitação chamada de Pregão, que pode ser na forma presencial ou eletrônica (preferencialmente).

Porém, existem casos em que um processo de licitação pode ser dispensado, ou inexigível. Isso ocorre quando um órgão público deseja realizar uma contratação, e esta contratação atende a algum dos critérios dispostos no art. 74 (inexigibilidade) ou art. 75 (dispensa), ambos da Lei 14.133/2021. Sendo assim, é possível a utilização desta forma de contratação. De maneira resumida, estas formas de contratação apresentam uma vantagem em relação as contratações por outras modalidades de licitação tradicionais, o tempo obrigatório entre a publicação dos editais e a definição do fornecedor vencedor é menor. Portanto, há uma vantagem quanto a diminuição no tempo para a contratação do fornecedor nestes casos. Por outro lado, estas contratações trazem menos disputa entre os fornecedores e o preço pode ficar elevado. Nestes casos, o gestor público deve levar em conta a real necessidade de utilização destas modalidades e também verificar se os preços do fornecedor estão de acordo com aqueles praticados no mercado.

Conforme visto anteriormente, a realização de uma Contratação Pública pode acontecer através de Licitação Pública ou através de contratação direta, sem uma Licitação Pública. Contudo, em ambos os casos, é possível dividir o processo de contratação em 3 (três) etapas, o planejamento, a execução e a contratação. Cada uma destas etapas será melhor detalhada nas sub-sessões abaixo.

### 4.2.1 Planejamento

Em praticamente todas as atividades profissionais, um planejamento adequado quase sempre traz benefícios para o andamento das atividades a que se deseja desempenhar para cumprir determinado objetivo. Em Instituições Públicas isso é ainda mais importante. As contratações de produtos e serviços, nesse caso, devem passar por todas as etapas de um processo de licitação pública.

Em uma contratação pública, podemos chamar de etapa de planejamento toda a fase interna de um processo licitatório. De acordo com artigo publicado por renomada empresa de consultoria na área, que pode ser visto em [Vianna & Consultores Associados Ltda \(2022\)](#), fazem parte da fase interna de um processo licitatório as seguintes etapas:

1. **Identificação da necessidade:** o administrador deve mostrar uma necessidade que precisa ser solucionada;
2. **Requisição do objeto:** a requisição deve ser elaborada pelo setor que necessita a aquisição dos produtos ou serviços. Ele deve ser claro e preciso. Deve conter uma justificativa para a aquisição e também deve ser apontada uma solução a que se deseja contratar.
3. **Autorização de abertura de procedimento licitatório:** a autoridade competente da instituição deve autorizar a abertura do processo licitatório para a referida contratação;
4. **Abertura de processo administrativo:** um processo administrativo deve ser aberto, devendo este ser autuado e numerado. Pode-se utilizar de ferramentas eletrônicas para isso.
5. **Elaboração de Projeto Básico/Termo de Referência:** um agente público que disponha de conhecimentos técnicos sobre o que se deseja contratar deve elaborar tal documento. Ele deve expressar detalhadamente as características do objeto a que se deseja contratar, bem como as obrigações que cabem a entidade contratante e a contratada para a prestação do serviço. Deve conter prazo e forma de execução, cronograma de pagamento, etc.
6. **Orçamento estimado:** uma pesquisa de mercado deve ser realizada com o objetivo de coletar orçamentos para o que se deseja contratar. É uma prática recomendada a utilização de pelo menos 3 (três) ou mais orçamentos. Após isso, um valor estimado para a contratação deve ser calculado, com base nos orçamentos apresentados.
7. **Previsão/indicação de recursos orçamentários:** normalmente, Órgãos Públicos possuem uma Lei Orçamentária Anual ([LOA](#)). Esta lei indica quanto e onde pode

gastar os seus recursos. Nele contém elemento e subelemento de despesa, que consistem em uma classificação dos recursos. Normalmente, setores de contabilidade e controle de execução orçamentária fornecem pareceres indicando a disponibilidade ou não de recursos e indicando elemento/subelemento de despesa para a referida contratação.

8. **Definição de modalidade e tipo de licitação:** As modalidades da licitação, pela Nova Lei de Licitações ([NLL](#)), são concorrência, pregão, leilão, concurso e diálogo competitivo. Porém, é possível também contratar através de dispensa/inexigibilidade de licitação, em situações específicas. Destaco neste caso a dispensa de licitação para contratações com outros Órgãos Públicos, conforme art. 75, inciso XIV, da Lei 14.133/2021, conforme segue:

“[...] para contratação de associação de pessoas com deficiência, sem fins lucrativos e de comprovada idoneidade, por órgão ou entidade da Administração Pública, para a prestação de serviços, desde que o preço contratado seja compatível com o praticado no mercado e os serviços contratados sejam prestados exclusivamente por pessoas com deficiência;”

Outra coisa importante de se destacar é a possibilidade de aderir a [ARPs](#) (Atas de Registro de Preços) de outras instituições que já realizaram processos licitatórios e que atendem a demanda necessária para a contratação. Isso pode facilitar o processo de contratação.

Os tipos de licitação a serem definidos podem ser, menor preço, melhor técnica e técnica e preço. Menor preço é quando o fornecedor do produto ou serviço puder ser definido utilizando-se como critério apenas o preço. Melhor técnica envolve serviços de natureza intelectual. Técnica e preço é uma mistura de ambas anteriores. Em todas elas, os critérios para classificar os fornecedores devem ser muito bem definidos.

Também deve ser definida a forma de julgamento dos itens a serem contratados. Pode ser menor preço por item, menor preço por lote ou menor preço global. Quando for menor preço por item, haverá um vencedor para cada item licitado. Menor preço por lote é quando um conjunto de itens é agrupado em um lote. Neste caso, deve-se justificar por que o agrupamento destes itens é relevante. Menor preço global é quando haverá apenas um fornecedor vencedor para toda a licitação. Também nesse caso deve-se justificar agrupamentos de itens.

9. **Elaboração e publicação de edital:** neste documento, todas as cláusulas para a contratação são estabelecidas para que os fornecedores concorrentes tomem conhecimento. Nele, devem conter itens indicando os documentos de habilitação, como será o julgamento de propostas, quais as especificações do objeto, prazos, etc. Após a completa elaboração do edital, o mesmo deve ser publicado no veículo de imprensa

oficial, no sítio eletrônico e nas redes sociais da instituição. Esta é a última etapa da fase interna, ou de planejamento de um processo de contratação em uma Entidade Pública.

Tamanha importância têm o planejamento em processos licitatórios que, a partir da [NLL](#), a Lei 14.133/2021, o [ETP](#) passou a ser obrigatório ao invés de opcional. Importante salientar também que a utilização da [NLL](#) passa a ser obrigatória a partir da data de 1º de abril de 2023.

O Tribunal de Contas da União ([TCU](#)) Também disponibiliza uma página em seu sítio contendo fluxogramas e orientações para as contratações públicas de todas as etapas de um processo de aquisição pública. Eles podem ser vistos em [Tribunal de Contas da União \(2022\)](#).

#### 4.2.2 Execução

Após concluídas todas as etapas da fase de planejamento de uma contratação pública, deve-se dar início a fase de execução da contratação, que podemos chamar também de fase externa. A Secretaria de Administração do Governo do Estado da Bahia ([SAEB](#)) através de documento disponível em [Governo do Estado da Bahia \(2018\)](#), classifica as etapas da fase externa da licitação da seguinte maneira:

1. **Convocatória:** consiste na divulgação do edital da licitação. Normalmente a divulgação do edital deve ocorrer no diário oficial eletrônico, no sítio eletrônico e nas redes sociais da instituição contratante. Também é comum a divulgação em jornais de grande circulação local e regional da sede da contratante. Conforme consta no art. 54, da [NLL](#), "*a publicidade do edital de licitação será realizada mediante divulgação e manutenção do inteiro teor do ato convocatório e de seus anexos no Portal Nacional de Contratações Públicas (PNCP)*"
2. **Classificatória:** em data, hora, local informados no edital da licitação, os fornecedores interessados em prestar os serviços deverão postar suas propostas, que deverão ser classificadas conforme critérios estabelecidos no edital de convocação.
3. **Habilitatória:** esta etapa consiste na análise de documentos exigidos no edital.
4. **Recursal:** quando algum fornecedor encontrar motivação para que alguma decisão nas etapas classificatória ou habilitatória seja revista, então este deve apresentar seus argumentos. A decisão pode ou não ser julgada procedente, levando a escolha de novo fornecedor. Normalmente, o primeiro julgador da peça recursal é o Agente de Contratação nomeado pela Instituição que realiza o processo de contratação. Se

houver discordância entre o julgador e o recurso apresentado, então a autoridade máxima da Instituição deve realizar novo julgamento.

5. **Homologatória e Adjudicatória:** após finalizados todos os atos praticados na fase externa e findados os prazos recursais, o processo licitatório deve ser homologado e adjudicado em favor do fornecedor que vencer a licitação. De forma resumida, isso consiste realizar o anúncio do vencedor da licitação e dar a garantia de que o produto ou serviço, caso sejam adquiridos, serão com o fornecedor vencedor.

### 4.2.3 Contratação

Após concluídas as fases interna e externa de uma licitação, pode-se iniciar a etapa de contratação. A etapa de contratação pode ser dividida em 3 (três) fases, a execução do contrato, recebimento do objeto e a liquidação das despesas.

- **Execução do contrato:** nesta etapa, um contrato ou instrumento equivalente deve ser firmado entre a empresa contratante e o fornecedor que venceu a licitação. Sendo assim, a empresa contratada e o Órgão Público contratante devem seguir fielmente todas as cláusulas presentes no documento assinado pelos representantes de ambas.
- **Recebimento do objeto:** Após assinado o contrato ou instrumento equivalente, o Órgão contratante deve emitir documento para que os produtos sejam entregues ou os serviços sejam iniciados. Cabe a entidade contratante fiscalizar o cumprimento destas cláusulas. Nesse caso, deve ser nomeado um fiscal para acompanhar a prestação dos serviços e verificação das especificações do que está sendo entregue. Além disso, o fiscal deverá efetuar notificações a empresa contratada caso alguma(s) cláusula(s) esteja(m) sendo descumprida(s).
- **Liquidação das despesas:** produtos entregues e serviços sendo executados sem nenhum problema apontado pelo fiscal do contrato, nos prazos e condições estabelecidas no contrato ou instrumento equivalente, deve ser realizado o pagamento para o fornecedor. Não havendo problemas, a Administração do Órgão Público contratante emite uma nota de empenho, que é uma espécie de "garantia" de que há orçamento previsto para o pagamento das despesas. Em seguida, deve ser realizada a liquidação da despesa. Para isso, as notas fiscais dos produtos ou serviços prestados devem ser apresentadas. Por último, o pagamento é efetivamente realizado para o contratado.



### 4.3 Tópicos relevantes para implantação/contratação de serviço em nuvem

Para facilitar a compreensão, o texto será dividido em tópicos relevantes a ser levados em consideração quando se deseja implantar um sistema em nuvem.

#### 4.3.1 Contratações de serviço em nuvem em outros órgãos públicos

Nesta sub-seção, serão apresentados 3 (três) casos, os quais servirão como exemplo de propostas de outros órgãos públicos para a contratação de serviços em nuvem. No primeiro caso, o Conselho de Arquitetura e Urbanismo de Santa Catarina ([CAU/SC](#)), no ano de 2020, realizou um CONTRATO DE ADESÃO DE PRESTAÇÃO DE SERVIÇOS ESPECIALIZADOS DE TECNOLOGIA DE INFORMAÇÃO, presente em [SERVIÇO FEDERAL DE PROCESSAMENTO DE DADOS – SERPRO \(2022\)](#), com o Serviço Federal de Processamento de Dados ([SERPRO](#)), para contratar um serviço de cloud do tipo [IaaS](#). O [SERPRO](#) trata-se de uma empresa Pública Federal com sede em Brasília/DF. Esta empresa oferece para outros Órgãos Públicos, além de um serviço de consultoria com diversos profissionais competentes na área de nuvem, mas também fornece o serviço de nuvem que seja o mais viável apontado pela consultoria realizada pela equipe do [SERPRO](#). Para um órgão público contratar com o [SERPRO](#), é necessário preencher um cadastro inicial presente em <https://campanhas.serpro.gov.br/serpro-multicloud/>. [IaaS](#) é um modelo de serviço que já foi brevemente explicado neste trabalho na sessão 2.4.2. Este contrato foi celebrado por dispensa de licitação com base no inciso XVI do artigo 24 da Lei nº 8.666 de 1993.

No segundo caso, a Prefeitura do Município de Pompeia, no estado de São Paulo, realizou pregão presencial para a CONTRATAÇÃO DE EMPRESA PARA PRESTAÇÃO DE SERVIÇOS EM CLOUD COMPUTING (COMPUTAÇÃO EM NUVEM), CONTEMPLANDO O FORNECIMENTO DE AMBIENTE VIRTUAL (INFRAESTRUTURA COMO SERVIÇO). Após a realização deste pregão, foi firmado o Contrato 43/2020 com o fornecedor vencedor. As especificações do serviço de nuvem contratados e seus valores encontram-se abaixo, na figura 11.

Conforme podemos observar na tabela 11, nesta situação, a instituição pública contratante optou pela aquisição de um fornecedor de serviço de nuvem do tipo [IaaS](#). Foram alugadas 7 (sete) máquinas virtuais com configurações distintas, variando de R\$ 515,69 (quinhentos e quinze reais e sessenta e nove centavos) a R\$ 1.161,40 (um mil, cento e sessenta e um reais e quarenta centavos) mensais. A vigência do contrato é de 12 (doze) meses, e pode ser renovado por iguais e sucessivos períodos, até um total de 48 (quarenta e oito) meses.

Figura 11 – Tabela de itens, descrição e valores - Contrato 43/2020 da Prefeitura de Pompeia.

Item	Descrição Sumária	Valor Unitário	Valor Total
1	Máquina Virtual para hospedar o "Banco de Dados" . Especificações: Processador: 8 vCPU; RAM: 16GB; SSD 500GB; Sistema Operacional: Ubuntu Server 16.04 64 Bits	R\$ 1.057,71	R\$ 12.692,52
2	Máquina Virtual para hospedar "Aplicação Web I" Especificações: Processador: 8 vCPU; RAM: 12GB; SSD 200GB; Sistema Operacional: Ubuntu Server 16.04 64 Bits	R\$ 874,28	R\$ 10.491,36
3	Máquina Virtual para hospedar Servidor de Arquivos Especificações: Processador: 4 vCPU; RAM: 8GB; SSD 1000GB; Sistema Operacional: Windows Server 2008 64 Bits.	R\$ 1.161,40	R\$ 13.936,80
4	Máquina Virtual para hospedar "Banco de Dados e Aplicação Web" Especificações: Processador: 4 vCPU; RAM: 8GB; SSD 200GB; Sistema Operacional: Windows Server 2008 64 Bits.	R\$ 629,82	R\$ 7.557,84
5	Máquina Virtual para hospedar "GLPI" (Gerenciamento de Chamados T.I.) Especificações: Processador: 4 vCPU; RAM: 8GB; SSD 200GB; S.O.: CentOS 7 64 Bits.	R\$ 515,69	R\$ 6.188,28
6	Máquina Virtual para hospedar Active Directory - Master Especificações: Processador: 4 vCPU; RAM: 8GB; SSD 200GB; Sistema Operacional: Windows Server 2016 64 Bits	R\$ 630,51	R\$ 7.556,12
7	Máquina Virtual para hospedar Active Directory – Slave Especificações: Processador: 4 vCPU; RAM: 8GB; SSD 200GB; Sistema Operacional: Windows Server 2016 64 Bits.	R\$ 630,51	R\$ 7.556,12
8	Link de Comunicação Privada de conexão entre o ambiente em nuvem (Datacenter) e o Paço Municipal com velocidade mínima de 500Mbps Full Duplex.	R\$ 1.168,41	R\$ 14.020,92
<b>Valor Total</b>			<b>R\$ 79.999,96</b>

No terceiro caso, o Tribunal Regional do Trabalho da 1ª Região - RJ (TRT1) optou por aderir a ARP 11/2021 do Ministério da Economia, citado no item 4.1.2 deste trabalho. E fez isso através do PROCESSO Nº 4801/2022. Então, foi firmado o Contrato nº 2022-0015. As quantidades de itens contratados neste caso são os seguintes:

Figura 12 – Tabela de itens, descrição e valores - Contrato nº 2022-0015 do TRIBUNAL REGIONAL DO TRABALHO DA 1ª REGIÃO

Item	CATSER	Descrição	Unidade	Quantidade	Valor Unitário R\$	Valor Total (24 meses)
01	26050	Serviços de Computação em nuvem – Infraestrutura como Serviço (IaaS)	Unidade de Serviço de Computação em Nuvem - USN	810.641	1,88	1.524.004,41
02	26069	Plataforma como Serviço - PaaS	Unidade de Serviço de Computação em Nuvem - USN	203.022	1,51	306.562,57
03	26077	Software como Serviço - SaaS	Unidade de Serviço de Computação em Nuvem - USN	45.040	0,62	27.924,80
04	27081	Serviço de Gerenciamento e Operação de recursos em nuvem	Unidade - Instância gerenciada por mês	1272	86,70	110.282,40
05	27081	Serviço de Migração de Recursos Computacionais	Unidade - Instância de Computação migrada	52	259,00	13.468,00
06	27081	Serviço de Migração de Banco de dados	Unidade - Instância de Banco de Dados migrada	1	255,00	255,00
07	3840	Treinamento	Unidade - Turma de treinamento	2	8.000,00	16.000,00

O que observa-se neste caso é que foi feito um levantamento da necessidade da Instituição. Então, foi elaborado um Termo de Referência que contém as especificações dos itens a serem contratados. O que se observa é que as especificações deste documento são similares a aquelas presentes na ARP 11/2021. Então foi firmado contrato com o fornecedor

vencedor do processo que deu origem a [ARP 11/2021](#), do Ministério da Economia, com as quantidades e itens solicitados pelo [TRT1](#). A vigência do contrato é de 24 (vinte e quatro) meses e pode ser prorrogado até um total de 60 (sessenta) meses.

Pode-se observar que há diferentes maneiras de contratar serviços em nuvem em instituições públicas, como visto acima. Uma delas é através de contrato com o [SERPRO](#), em processo de dispensa de licitação. Outra é através de adesão a [ARP 11/2021](#) do Ministério da Economia. E a outra opção é a instituição realizar um processo de licitação para contratar o serviço no formato que julgar necessário.

## 4.4 Equipe de profissionais na área disponíveis

Algo importante a ser levado em consideração, quando escolher entre contratação de infraestrutura em nuvem de terceiros ou montar e manter operante todo o parque tecnológico de uma instituição, é a quantidade de profissionais que dispõe para manter estes serviços. Quando falamos de computação em nuvem, podemos separar em dois grandes grupos de funções e tarefas que deverão ser realizadas pelos profissionais que dispõe no plantel da instituição ([ANHANGUERA, 2021](#)).

O primeiro deles está relacionado com dar manutenção a servidores em nuvem e desenvolver os softwares a serem hospedados em nuvem. O profissional que realizar estas tarefas deve apresentar um perfil analítico e deve ser capaz de resolver problemas complexos. Existem dois cargos relevantes neste grupo, o de arquiteto de nuvem e arquiteto de segurança da informação.

Um segundo grupo possui funções que atuam diretamente nas áreas de *Big Data*, *Internet of Things* ([IoT](#)), inteligência artificial e [HPC](#).

Ocorre que os profissionais nas áreas citadas em ambos os casos são melhor remunerados no setor privado, o que acaba atraindo os melhores profissionais para estas áreas. Outro problema é que as tecnologias relacionadas a computação em nuvem são recentes e novas tecnologias surgem em pouco tempo. E como é comum que um trabalhador entre para o serviço público e por lá permaneça por muitos anos, é importante realizar capacitações frequentes para requalificar um servidor público para atuar nestas áreas. Outro problema a ser levado em consideração é que o servidor público deve obedecer aos deveres e as normas de conduta para o cargo a que ele prestou concurso, conforme diz o art. 2º, inciso I, da Lei 8.027/1990, conforme segue:

*Art. 2º São deveres dos servidores públicos civis:*

*I - exercer com zelo e dedicação as atribuições legais e regulamentares inerentes ao cargo ou função;*

Ou seja, o servidor público deve executar apenas as atribuições do seu cargo. Porém,

é possível conceder novas atribuições a um servidor através de funções gratificadas. As funções gratificadas consistem em retribuições relativas ao exercício de função de direção, chefia, assessoramento, entre outros, o qual é acrescido um valor fixo no vencimento do servidor. Cada Instituição Pública possui regulamentações diferentes relativas a concessão de gratificações e funções gratificadas a seus servidores, e isso deve ser analisado pelo responsável pelo recrutamento de pessoal para os serviços em nuvem a que se refere este trabalho.

Como visto no parágrafo anterior, pode ser complicado dispor de profissionais para atuar na área de computação em nuvem. Uma maneira de resolver este problema pode ser através de locação de infraestrutura de outras empresas ou contratação de serviços, que podem incluir boa parte das tarefas necessárias para a prestação de serviço de computação em nuvem, inclusive a mão-de-obra. Este trabalho traz alguns pontos importantes para contratações em Órgãos Públicos na sessão [4.2](#).

## 4.5 Infraestrutura própria *versus* alugada

A comparação entre infraestrutura própria e alugada, na verdade, pode ser traduzida para uma comparação entre uma nuvem privada ou nuvem pública. O objetivo deste trabalho não é realizar uma análise comparativa com os custos entre a implantação destes tipos de nuvens, uma vez que outros trabalhos já fazem este tipo de análise, como é o caso de [PATRICIO \(2018\)](#). Este autor realizou uma análise de todos os custos envolvidos na implantação de uma nuvem pública, privada e híbrida com base em uma solução utilizando *Microsoft Azure*. No caso em questão, o cenário de uma empresa X é utilizado como exemplo. Os custos para manter em uma nuvem privada os serviços da empresa são levantados. Em seguida, é feito um levantamento para migrar a nuvem privada para uma nuvem pública. Um outro cenário de migração é feito, migrando apenas alguns serviços para nuvem pública e mantendo outros em nuvem privada, caracterizando assim um cenário de nuvem híbrida. Ao final, o custo mensal nestes 3 (três) cenários ficou mais baixo em um cenário de nuvem privada. Porém, o autor deixa claro em suas considerações finais que “pode-se concluir que não existe um modelo ideal, que isto sempre dependerá da aplicação, sistema, ambiente e serviço”. Também importante dizer que, no caso analisado, as aplicações em nuvem não foram desenvolvidas para serem processadas em nuvem pública e, não sendo compatíveis com escalabilidade e elasticidade, o que torna os custos em nuvens pública e híbrida elevados.

Como visto no capítulo anterior, a escolha entre manter uma nuvem privada ou migrar para uma nuvem pública depende muito do cenário a que se estiver analisando. Nuvens públicas costumam ser muito mais baratas quando são projetadas para operar com elasticidade e escaláveis.

Algo que também deve ser levado em consideração é que pode haver uma infraestrutura legada na instituição que se está analisando. Neste caso, é possível manter parte dos serviços em nuvem privada e migrar outros para nuvem pública. O importante é analisar os cenários possíveis, fazer um levantamento de custos e então realizar a tomada da melhor decisão para o caso em questão. Deve-se tomar precauções quanto a possíveis aumentos de demanda na nuvem. Custos para ampliar a infraestrutura de uma nuvem privada podem ser muito grandes a partir de um certo ponto. Já em uma nuvem pública, é possível expandir a infraestrutura temporariamente, aumentando seu espaço, poder de processamento e tráfego de rede por determinados períodos. Isso com um custo relativamente pequeno se comparado com aquisição de novos equipamentos em uma nuvem privada na maioria do tempo apresentando uma nuvem subutilizada.

Questões como mão-de-obra, treinamentos de pessoal, requisitos dos produtos/serviços que necessitam ser comprados/contratados precisam ser criteriosamente verificados. Verificar os melhores preços que o mercado tem para oferecer também é relevante para reduzir os custos, de preferência com diversos fornecedores. Se a opção for pela contratação de serviço de nuvem, verificar qual contratação atende melhor a demanda ([IaaS](#), [PaaS](#) ou [SaaS](#)).

#### 4.5.1 Contratação padronizada

Normalmente, empresas de tecnologia podem fornecer serviços como nuvem pública, onde todos os usuários podem acessar seu conteúdo de maneira padronizada. É comum esse tipo de serviço apresentar soluções unificadas para diversos clientes distintos, e então cada cliente utiliza sua solução da maneira como bem entender. Dessa forma, soluções deste tipo são economicamente mais baratas.

#### 4.5.2 Contratação personalizada

Quando uma instituição apresenta um cenário em que o serviço de nuvem pública fornecido pelas empresas a um preço mais baixo não atende a demanda, então um novo tipo de solução deve ser adotado. Geralmente isso ocorre quando há algumas funções em alguma infraestrutura legada, mas que não podem ser padronizadas ou apresentam processos voltados para as regras do contratante.

Conforme verificamos na tabela de preços presentes na [ARP 11/2021](#), presente na [figura 10](#) deste trabalho, pode-se concluir que um [SaaS](#) costuma ser a opção mais barata. Em seguida, um [PaaS](#) e depois um [IaaS](#). Ou seja, quanto menor o nível de controle por parte do usuário, menor o custo de operação. Portanto, o ideal é que as aplicações sejam modeladas para operarem, de preferência, e se possível é claro, através de um [SaaS](#). A segunda melhor opção seria um [PaaS](#), e por último, um [IaaS](#).

Porém, quando há uma infraestrutura legada, com utilização de uma nuvem privada, mas se deseja migrar para uma nuvem pública, ou abordar uma solução com nuvem híbrida, é necessário ter uma preocupação extra com a migração dos dados e com o suporte personalizado na hora de contratar um provedor de serviços em nuvem.

## 5 Conclusões

Não existe apenas um cenário ou uma tecnologia que resolva todos os problemas de qualquer instituição. Cada local apresenta características diferentes. O importante é que alguém que deverá tomar decisões sobre aquisição de serviços em nuvem tenha conhecimento do cenário existente em sua instituição, dos rumos que serão tomados futuramente e também o que o mercado tem a oferecer como solução para as demandas exigidas. Outro fator que dificulta a atuação para solucionar problemas em órgãos públicos é a enorme quantidade de leis, instruções normativas e outras regras necessárias de serem cumpridas para conseguir realizar uma contratação.

Este trabalho apresentou 3 (três) maneiras para um órgão público realizar a aquisição de serviços em nuvem. A primeira delas é referente a adesão a [ARP 11/2021](#), conforme visto no item [4.1.2](#) deste trabalho. A adesão a esta [ARP](#) é mais simples do que se fosse necessário montar todo um processo licitatório independente, seguindo todos os passos mostrados no item [4.2](#) deste trabalho. Certamente seria algo mais burocrático e demorado. O ideal é que se verifique se o cenário atual e futuro de quem deseja contratar serviço em nuvem se enquadra nos itens contratados na [ARP 11/2021](#), ou em outra contratação que o Governo Federal venha a realizar futuramente, desde que seja possível aderir a esta outra [ARP](#). A segunda maneira foi através do [SERPRO](#), conforme exemplo apresentado no item [4.3.1](#) deste trabalho. Conforme visto, é possível contratar este outro órgão público para fornecer uma consultoria sobre qual serviço de nuvem seria o mais adequado para o contratante e também fornece tais serviços.

Estas duas primeiras opções apresentadas nos dois parágrafos anteriores servem para contratar uma empresa que forneça o serviço de nuvem para o contratante. Neste caso, haverá uma preocupação menor com mão-de-obra por parte da contratante, pois a implantação, migração e manutenção da nuvem fica por conta do contratado. Porém, a terceira maneira de realizar aquisição de serviço de nuvem seria para quando a instituição pública decidir implantar sua própria nuvem privada. Neste caso, seriam necessários diversos profissionais para mantê-la operante. E conforme já visto no item [4.4](#) deste trabalho, estes profissionais são caros, o setor privado é mais atrativo financeiramente e contratar os profissionais necessários é difícil para os órgãos públicos.

Certamente, decidir a melhor maneira para adquirir uma nuvem é uma escolha difícil. Um estudo técnico deve ser realizado para definir qual a opção mais viável, contratar um serviço de nuvem ou implantar infraestrutura própria. Caso os estudos constatarem que o mais viável seja a aquisição de nuvem privada, é importante tomar alguns cuidados. É importante planejar a solução para que em um futuro, caso a demanda por nuvem



diminua, outra solução mais barata possa ser adquirida. Ou por qualquer outro motivo se decida abandonar a nuvem privada, que a migração para nuvem pública seja possível. Por isso, algo importante em se preocupar desde cedo é utilizar a cultura DevOps, conforme visto no item 2.5 deste trabalho.

Outro ponto importante é referente a utilização da cultura DevOps, que é uma ótima prática para melhorar processos quando se fala em computação em nuvem. Investimentos em treinamento dos profissionais que atuam nesta área fazem diferença para melhorar a eficiência dos profissionais que atuam nos serviço de nuvem. Outros investimentos em capacitação também são importantes para manter as equipes sempre atualizadas, pois assuntos tecnológicos apresentam mudanças constantes em curtos períodos de tempo.

Por fim, importante que pessoas que tomam decisões sobre contratações de serviço em nuvem em órgãos públicos fiquem sempre atentos a atualizações em leis ou normas internas nas instituições a que fazem parte. Pois frequentemente há mudanças nestas áreas. Uma delas que quero citar neste momento é a [NLL](#), que entra em vigor obrigatoriamente em 1º de abril de 2023. Outras [ARPs](#) disponibilizadas por outros órgãos públicos da esfera Federal possíveis de serem aderidas.

E o mais importante em qualquer uma das situações, realizar um estudos técnicos e planejamento com cautela. Quanto mais informações forem levantadas nesses documentos, melhor será a escolha. E tomar cuidado para que a pressa em implantar logo uma mudança não afete todo o andamento de um projeto.

# Referências

- Advanced Micro Devices. *What High Performance Computing Solutions can do for you*. 2022. Acessado em: 06 de setembro 2022. Disponível em: <<https://www.xilinx.com/applications/data-center/high-performance-computing.html>>. Citado na página 38.
- Altera Corporation. *Accelating High-Performance Computing With HPGAs*. 2007. Acessado em: 06 de setembro 2022. Disponível em: <<https://www.intel.com/content/dam/support/us/en/programmable/support-resources/bulk-container/pdfs/literature/wp/wp-01029.pdf>>. Citado na página 38.
- AMAZON. *Amazon S3*. 2022. Acessado em: 28 de setembro 2022. Disponível em: <<https://aws.amazon.com/pt/s3/?nc=sn&loc=0>>. Citado na página 45.
- AMAZON. *Computação de alta performance - Acelere a inovação com redes rápidas e infraestrutura virtualmente ilimitada*. 2022. Acessado em: 22 de agosto 2022. Disponível em: <<https://aws.amazon.com/pt/hpc/>>. Citado na página 38.
- ANHANGUERA. *Cloud Computing: o que é e como é a atuação na área?* 2021. Url<https://blog.anhanguera.com/cloud-computing/>. Citado na página 59.
- ARAÚJO, C. C. L. *COMPUTAÇÃO EM NUVEM: um estudo sobre a distribuição da produção de artigos publicados no período de 2007 a 2016*. 2017. Citado na página 16.
- AUTHORS, R. *What is Rook?* 2023. Acessado em: 10 de Novembro 2021. Disponível em: <<https://rook.io/>>. Citado na página 31.
- BOTOLE, M. P. *Mainframe e Atualidade*. 2011. Citado na página 20.
- BUFFA, L. H. *TORNANDO ACESSÍVEL A CULTURA DEVOPS A PEQUENAS EMPRESAS E STARTUPS*. 2021. Citado na página 27.
- Ceph Documentation. *ARCHITECTURE - Ceph Documentation*. 2016. Acessado em: 10 de Novembro 2021. Disponível em: <<https://docs.ceph.com/en/latest/architecture/>>. Citado na página 31.
- CLARO, C. A. M. *PERSISTÊNCIA POLIGLOTA EM BANCOS DE DADOS E FERRAMENTAS DE DESENVOLVIMENTO DE APLICATIVOS*. 2017. Acessado em: 20 de setembro 2022. Disponível em: <[http://repositorio.utfpr.edu.br:8080/jspui/bitstream/1/19842/1/CT\\_TECOSOL\\_II\\_2017\\_01.pdf](http://repositorio.utfpr.edu.br:8080/jspui/bitstream/1/19842/1/CT_TECOSOL_II_2017_01.pdf)>. Citado na página 42.
- DEITEL, H. M.; DEITEL, P. J.; CHOFFNES, D. R. *Sistemas Operacionais*. 3. ed. [S.l.]: ed. São Paulo: Pearson Prentice Hal, 2005. Citado na página 22.
- DEVICES, A. M. *Explicação da computação de alto desempenho*. 2022. Acessado em: 06 de setembro 2022. Disponível em: <<https://www.amd.com/pt/technologies/hpc-explained>>. Citado na página 38.

DOCKER. *Docker makes development efficient and predictable*. 2022. Acessado em: 22 de junho 2022. Disponível em: <<https://www.docker.com/>>. Citado na página 29.

Gaea Consulting. *Benefícios do DevOps conheça os 5 principais*. 2022. Acessado em: 09 de Dezembro de 2022. Disponível em: <<https://gaea.com.br/beneficios-do-devops-conheca-os-5-principais/>>. Citado na página 27.

GOOGLE. *Cloud Storage*. 2022. Acessado em: 28 de setembro 2022. Disponível em: <<https://cloud.google.com/storage?hl=pt-br>>. Citado na página 45.

Governo do Estado da Bahia. *MÓDULO II - FASES INTERNA E EXTERNA DO PROCEDIMENTO LICITATÓRIO*. 2018. Acessado em: 07 de Novembro de 2022. Disponível em: <[https://comprasnet.ba.gov.br/sites/default/files/modulo\\_ii\\_-\\_fases\\_interna\\_e\\_externa\\_do\\_procedimento\\_licitatorio\\_0.pdf](https://comprasnet.ba.gov.br/sites/default/files/modulo_ii_-_fases_interna_e_externa_do_procedimento_licitatorio_0.pdf)>. Citado na página 54.

IBM. *O que é etcd?* 2023. Url<https://www.ibm.com/br-pt/topics/etcd>. Citado na página 33.

Instituto Federal de Santa Catarina – Reitoria. *Plano Diretor de Tecnologia da Informação e Comunicação – PDTIC 2021 – 2022*. 2021. Acessado em: 09 de Novembro de 2022. Disponível em: <<https://dtic.ifsc.edu.br/documentos/pdtic>>. Citado na página 48.

LEONE, L. de. *O que é DevOps? Processo, origem e problemática!* 2022. Url<https://becode.com.br/o-que-e-devops>. Citado na página 27.

MICROSOFT. *Azure Blob Storage*. 2022. Acessado em: 28 de setembro 2022. Disponível em: <<https://azure.microsoft.com/en-us/products/storage/blobs/#overview>>. Citado na página 45.

MICROSOFT. *Azure high-performance computing - Unlock your innovation with the leading HPC cloud solution*. 2022. Acessado em: 22 de agosto 2022. Disponível em: <<https://azure.microsoft.com/en-us/solutions/high-performance-computing/#intro>>. Citado na página 38.

Ministério da Economia. *Ata nº 11/2021 - Serviço de Computação em Nuvem*. 2021. Acessado em: 17 de Novembro de 2022. Disponível em: <<https://www.gov.br/economia/pt-br/assuntos/gestao/central-de-compras/transparencia/arp/2021/ata-ndeg-11-2021-servico-de-computacao-em-nuvem>>. Citado na página 50.

Ministério da Gestão e da Inovação em Serviços. *Estratégia de Governo Digital 2020-2022*. 2022. Acessado em: 26 de Outubro de 2022. Disponível em: <<https://www.gov.br/governodigital/pt-br/EGD2020>>. Citado na página 47.

Ministério do Planejamento, Desenvolvimento e Gestão. *Padrões de Interoperabilidade de Governo Eletrônico Documento de Referência*. 2018. Acessado em: 26 de Outubro de 2022. Disponível em: <[https://www.gov.br/governodigital/pt-br/governanca-de-dados/ePING\\_v2018\\_20171205.pdf](https://www.gov.br/governodigital/pt-br/governanca-de-dados/ePING_v2018_20171205.pdf)>. Citado na página 48.

Ministério do planejamento, orçamento e gestão. *Boas práticas, orientações e vedações para contratação de Serviços de Computação em Nuvem*. 2022. Acessado em: 29 de Outubro de 2022. Disponível em: <<https://www.gov.br/governodigital/pt-br/contratacoes/Orientacaoservicosemnuvem.pdf>>. Citado na página 48.

- Ministério do Planejamento, Orçamento e Gestão. *Guia Livre: Referência de Migração para Software Livre do Governo Federal*. 2022. Acessado em: 26 de Outubro de 2022. Disponível em: <[http://www.inmetro.gov.br/portaibrmlq/documentos\\_disponiveis/Guia\\_Livre\\_Ipiranga\\_v095.pdf](http://www.inmetro.gov.br/portaibrmlq/documentos_disponiveis/Guia_Livre_Ipiranga_v095.pdf)>. Citado na página 48.
- NIST. *The NIST Definition of Cloud Computing*. 2011. Acessado em: 21 de Novembro 2021. Disponível em: <<https://csrc.nist.gov/publications/detail/sp/800-145/final>>. Citado 3 vezes nas páginas 24, 25 e 26.
- NVIDIA. *Computação de Alto Desempenho*. 2022. Acessado em: 06 de setembro 2022. Disponível em: <<https://www.nvidia.com/pt-br/high-performance-computing/>>. Citado na página 38.
- ORACLE. *Computação de alto desempenho (HPC)*. 2022. Acessado em: 22 de agosto 2022. Disponível em: <<https://www.oracle.com/br/cloud/hpc/>>. Citado na página 38.
- PATRICIO, D. *UMA TOMADA DE DECISAO NA EMPRESA X: COMPARAÇÃO DOS CUSTOS ALOCADOS PARA A MANUTENÇÃO DO AMBIENTE COMPUTACIONAL EM NUVEM*. 2018. Acessado em: 29 de Novembro de 2022. Citado na página 60.
- RAMAKRISHNAN, R.; GEHRKE, J. *Sistema de Gerenciamento de Banco de Dados*. 3. ed. [S.l.]: Ed. New York: Mcgraw-hill, 2011. Citado na página 39.
- RAMOS, T. D. P. *ALTA DISPONIBILIDADE E BALANCEAMENTO DE CARGA PARA SERVIDORES WEB USANDO LINUX*. 2009. Acessado em: 28 de setembro 2022. Disponível em: <[http://repositorio.ufla.br/jspui/bitstream/1/5260/1/MONOGRAFIA\\_Alta\\_disponibilidade\\_e\\_balanceamento\\_carga\\_servidores\\_web\\_linux.pdf](http://repositorio.ufla.br/jspui/bitstream/1/5260/1/MONOGRAFIA_Alta_disponibilidade_e_balanceamento_carga_servidores_web_linux.pdf)>. Citado 3 vezes nas páginas 43, 44 e 45.
- RAVANÊDA, R. *COMPUTAÇÃO EM NUVEM E IDENTIFICAÇÃO, ORDENAMENTO E ANÁLISE DOS PROBLEMAS ASSOCIADOS À SUA ADOÇÃO*. 2014. Citado 2 vezes nas páginas 23 e 34.
- Red Hat. *O que é Kubernetes?* 2021. Acessado em: 14 de Novembro 2021. Disponível em: <<https://www.redhat.com/pt-br/topics/containers/what-is-kubernetes>>. Citado na página 32.
- REIS, R. B. *Teste de desempenho de sistemas de arquivos distribuídos em nuvem computacional privada no IFSC câmpus São José*. 2017. Citado 2 vezes nas páginas 30 e 31.
- RODRIGUES, M. *Computação em nuvem: Estudo de Viabilidade*. 2011. Citado na página 39.
- SERVIÇO FEDERAL DE PROCESSAMENTO DE DADOS – SERPRO. *CONTRATO DE ADESÃO DE PRESTAÇÃO DE SERVIÇOS ESPECIALIZADOS DE TECNOLOGIA DE INFORMAÇÃO – SERPRO CLOUD - Iaa S*. 2022. Acessado em: 20 de outubro de 2022. Disponível em: <<https://www.causc.gov.br/wp-content/uploads/contratos/28222/Contrato-assinado.pdf>>. Citado na página 56.
- SILVA, M. A. da. *Utilização da Computação Distribuída para o armazenamento e indexação de dados forenses*. 2012. Citado na página 22.

- SILVA, R. F. da. *Virtualização de Sistemas Operacionais*. 2007. Citado na página 30.
- SoftwareOne Brasil. *O que é um serviço de Cloud Broker?* 2020. Acessado em: 17 de Novembro de 2022. Disponível em: <<https://www.softwareone.com/pt-br/blog/artigos/2020/02/06/cloud-broker>>. Citado na página 49.
- SOUSA, A. S. D. *OS DESAFIOS DA COMPUTAÇÃO EM NUVEM – ESTADO DA ARTE*. 2014. Citado 3 vezes nas páginas 16, 26 e 29.
- SOUSA, A. S. D. *OS DESAFIOS DA COMPUTAÇÃO EM NUVEM – ESTADO DA ARTE*. 2014. Citado na página 37.
- SOUZA, G. D. *Nuvem privada multizona usando orquestração de contêineres: Proposta multicampus de alta disponibilidade para o IFSC*. 2019. Citado 2 vezes nas páginas 30 e 47.
- SOUZA, H. N. *PRINCIPAIS CONCEITOS DE SOA E SUA IMPLEMENTAÇÃO ATRAVÉS DE WEB-SERVICES*. 2011. Citado na página 38.
- TANENBAUM, A. S.; STEEN, M. V. *Sistemas distribuídos: princípios e paradigma*. 2. ed. [S.l.]: ed. São Paulo: Pearson Prentice Hall, 2007. Citado na página 22.
- TENORIO, R. B. D. A. *UMA PROPOSTA PARA USO DE CACHE DISTRIBUÍDO NO CONTEXTO DO TCE-RN*. 2022. Acessado em: 14 de setembro 2022. Disponível em: <[https://repositorio.ufrn.br/bitstream/123456789/48324/4/tcc\\_ricardo\\_tenorio\\_.pdf](https://repositorio.ufrn.br/bitstream/123456789/48324/4/tcc_ricardo_tenorio_.pdf)>. Citado na página 40.
- The Kubernetes Authors. *Kubernetes - Pods*. 2021. Acessado em: 14 de Novembro 2021. Disponível em: <<https://kubernetes.io/docs/concepts/workloads/pods/>>. Citado na página 32.
- The Kubernetes Authors. *Kubernetes Components*. 2021. Acessado em: 14 de Novembro 2021. Disponível em: <<https://kubernetes.io/docs/concepts/overview/components/>>. Citado na página 33.
- The Kubernetes Authors. *Kubernetes Documentation - API Overview*. 2021. Acessado em: 14 de Novembro 2021. Disponível em: <<https://kubernetes.io/docs/reference/using-api/>>. Citado na página 34.
- The Kubernetes Authors. *Production-Grade Container Orchestration - Kubernetes*. 2021. Acessado em: 14 de Novembro 2021. Disponível em: <<https://kubernetes.io/>>. Citado na página 32.
- Tribunal de Contas da União. *Aquisições públicas*. 2022. Acessado em: 03 de Novembro de 2022. Disponível em: <<https://www.tcu.gov.br/arquivosrca/>>. Citado na página 54.
- VIANA, A. S. D. B. *UTILIZAÇÃO DE UMA ARQUITETURA DE SOFTWARE BASEADA EM MICRO SERVIÇOS EM UM SISTEMA DE VENDAS ONLINE*. 2018. Citado na página 20.
- Vianna & Consultores Associados Ltda. *Planejamento das licitações*. 2022. Acessado em: 03 de Novembro de 2022. Disponível em: <<https://www.viannaconsultores.com.br/planejamento-das-licita%C3%A7%C3%B5es>>. Citado na página 52.