

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
PROGRAMA DE PÓS-GRADUAÇÃO EM TECNOLOGIA  
ESPECIALIZAÇÃO EM GESTÃO DE TECNOLOGIA DA INFORMAÇÃO E  
COMUNICAÇÃO

FABIANO CESAR KLEIN

**“UTILIZAÇÃO DE SISTEMA OPERACIONAL ARMAZENADO EM  
MEMÓRIA SÓLIDA PORTÁTIL NÃO-VOLÁTIL”**

MONOGRAFIA DE ESPECIALIZAÇÃO

CURITIBA

2015

FABIANO CESAR KLEIN

**“UTILIZAÇÃO DE SISTEMA OPERACIONAL ARMAZENADO EM  
MEMÓRIA SÓLIDA PORTÁTIL NÃO-VOLÁTIL”**

Monografia do Curso de Especialização em Gestão de Tecnologia da Informação e Comunicação da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Especialista em Gestão de Tecnologia da Informação e Comunicação.

Orientador: Prof. M.Sc. Alexandre Jorge Miziara

CURITIBA

2015

## DEDICATÓRIA

À minha mãe, Elita, (*in memoriam*).

## **AGRADECIMENTOS**

À Companhia Paranaense de Energia - COPEL, por reembolsar parte significativa dos custos destes estudos.

Aos estudantes e colegas, sem os quais, não formaríamos uma turma para realizar esta especialização.

À todos os professores e profissionais que participaram direta e ou indiretamente dos processos deste curso.

Ao Prof. D.Sc. Kleber Kendy Horikawa Nabas, pela sua experiência e maneira objetiva de conduzir os cursos de extensão e de especialização, os quais participei.

Ao Prof. D.Sc Carlos Alberto Maziero, por desempenhar esforços tremendos na área do conhecimento, envolvendo sistemas operacionais, estes, os quais tanto simpatizo.

Um forte agradecimento ao Prof. M.Sc. Alexandre Jorge Miziara, por coordenar o curso e a todos os envolvidos de uma maneira sem igual, sempre acreditando no aprendizado, mantendo a paciência, incentivando, passando a sua tranquilidade nos momentos mais difíceis, mostrando uma visão diferente e de longo alcance nos mais variados aspectos.

Por final, um agradecimento especial ao meu pai Sr. Eniomar Arnóbio Klein, o qual, sempre esteve e permanece ao meu lado nas batalhas diárias, por ter acreditado no meu potencial, proporcionando a aquisição do meu primeiro computador de mesa, em uma época em que a interface gráfica começava a ser explorada e a internet ainda era discada.

## LISTA DE ILUSTRAÇÕES

Figura1. ESTRUTURA DE UM SISTEMA DE COMPUTAÇÃO TÍPICO.....	20
Figura2. FUNCIONALIDADES DO SISTEMA OPERACIONAL.....	33
Figura3. ESTRUTURA DE UM SISTEMA OPERACIONAL.....	34

## LISTA DE TABELAS

Tabela1. PRIMEIRA PERGUNTA DA PESQUISA EMPÍRICA.....	17
Tabela2. SEGUNDA PERGUNTA DA PESQUISA EMPÍRICA.....	18
Tabela3. TERCEIRA PERGUNTA DAPESQUISA EMPÍRICA.....	19

## LISTA DE GRÁFICOS

Gráfico1. EXPOSIÇÃO DOS DADOS – PESQUISA EMPÍRICA.....	17
Gráfico2. EXPOSIÇÃO DOS DADOS – PESQUISA EMPÍRICA.....	18
Gráfico3. EXPOSIÇÃO DOS DADOS – PESQUISA EMPÍRICA.....	19

## RESUMO

KLEIN, Fabiano Cesar. Utilização de sistema operacional armazenado em memória sólida portátil não-volátil. 2015. Monografia (Especialização em Gestão de Tecnologia da Informação e Comunicação) – Diretoria de Pesquisa e Pós-Graduação, Universidade Tecnológica Federal do Paraná. Curitiba, 2015.

Com o paradigma: No mundo da conectividade e comunicação, cada vez mais compartilhado tecnologicamente, conseguimos imaginar: uma memória sólida portátil não-volátil, exemplo, um pen drive sem um sistema operacional pronto para ser utilizado, o qual pode rodar a qualquer momento, em qualquer lugar, nos mais variados equipamentos? Pouco sabemos quando vamos necessitar de tal acesso, porém, tê-lo em mãos e em qualquer lugar, e ainda com a opção de utilizar, testar e ou instalar o mesmo de forma segura, econômica e eficiente, o torna indispensável. Chegamos a pergunta problema: Como ter em mãos, há qualquer momento, em qualquer lugar, sistemas operacionais rodando nos mais diversos equipamentos, desempenhando várias atividades de operações, de maneira segura, econômica e eficiente, e o melhor de modo portátil? Neste trabalho acadêmico realizamos estudos, testes e pesquisas envolvendo soluções completas de sistemas operacionais armazenados em memórias portáteis. Detalhamos um teste em forma de passo-a-passo, envolvendo um sistema operacional livre armazenado em um pen drive, bem como pesquisa qualitativa com o objetivo de localizar e explorar usuários que utilizaram, utilizam e ou virão a utilizar tais soluções. Buscamos indivíduos que possuem um pen drive ou algum outro dispositivo de armazenamento envolvendo memória sólida portátil não volátil e quais destes indivíduos possuem no mesmo um sistema operacional pronto para ser utilizado. Entrevistas e questionários foram aplicados em indivíduos das mais variadas áreas, as quais estão conectadas de maneira direta e ou indireta a área de tecnologia da informação e comunicação (TIC). Como impacto e resultado nos colocamos como referência no assunto para novos estudos relativos à convergência tecnológica e dos meios de comunicação, bem como, uma possível base para patente na área.

**Palavras-Chave:** Sistema operacional, Boot via memória sólida portátil não-volátil.



## ABSTRACT

KLEIN, Fabiano Cesar. Operational system use stored in non-volatile solid portable memory. 2015. Monograph (Specialization in Information and Communication Technology Management) - Directorate of Research and Graduate Studies, Federal Technological University of Paraná. Curitiba, 2015.

With the paradigm: In the world of connectivity and communication, increasingly technologically shared, could we imagine: a non-volatile solid portable memory, such as USB flash drive without an operating system ready to be used, which can run at anytime, anywhere, in a variety of equipments? We don't know when we will need such access, however, having it in hands anywhere and even more, with the option to use, test and install it in a safe, economic and efficient way, makes it indispensable. We got to the problem question: How to have, in hands, any time, in any place, operational systems running in most diverse equipments performing several operational activities in a safe, economic and efficient way, and the best in a portable way? This academic work carried out studies, tests and researches involving complete solutions in operational system stored in portable memory. This test is detailed step by step, involving a free operational system stored in a USB flash drive as well as qualitative research in order to locate and explore users who used, use or will use such solutions. We sought for individuals who have a USB flash drive or any other storage device involving SSD portable non-volatile and which of these individuals have the same operational system ready to be used. Interviews and questionnaires were administered to individuals from varied areas, which are connected directly and or indirectly to information and communication technology area (ICT). As impact and results, we are set as a reference on the subject for further studies related to technological convergence of media, as well as a possible basis for a patent in the area.

**Keywords:** Operating System, Boot via Portable non-volatile solid memory.

## SUMÁRIO

1.	<b>INTRODUÇÃO.....</b>	<b>11</b>
2.	<b>PROBLEMATIZAÇÃO.....</b>	<b>12</b>
	<b>2.1 OBJETIVOS.....</b>	<b>12</b>
	2.1.1 OBJETIVOS GERAIS.....	12
	2.1.2 OBJETIVOS ESPECÍFICOS.....	12
	<b>2.2 DELIMITAÇÕES.....</b>	<b>13</b>
	<b>2.3 METODOLOGIA.....</b>	<b>13</b>
3.	<b>DESENVOLVIMENTO.....</b>	<b>14</b>
	<b>3.1 CAPÍTULO I (TUTORIAL PASSO-A-PASSO) .....</b>	<b>14</b>
	<b>3.2 CAPÍTULO II (PESQUISA EMPÍRICA) .....</b>	<b>17</b>
	<b>3.3 CAPÍTULO III (FUNDAMENTAÇÃO TEÓRICA) .....</b>	<b>20</b>
	3.3.1 HISTÓRIA DOS SISTEMAS OPERACIONAIS.....	21
	3.3.2 SISTEMA OPERACIONAL.....	22
	3.3.3 TIPOS DE SISTEMAS OPERACIONAIS.....	25
	3.3.3.1 FUNCIONALIDADES.....	33
	3.3.4 ESTRUTURA DE UM SISTEMA OPERACIONAL.....	33
	3.3.5 DEFINIÇÕES E NOTAÇÃO DE ARMAZENAMENTO.....	35
	3.3.6 O ESTUDO DOS SISTEMAS OPERACIONAIS.....	35
	3.3.7 SISTEMA DE ARQUIVOS.....	36
	3.3.8 GERENCIAMENTO DE MEMÓRIA.....	38
	3.3.8.1 MEMÓRIA PORTÁTIL NÃO-VOLÁTIL.....	40
4.	<b>ANÁLISE DOS DADOS.....</b>	<b>41</b>
5.	<b>CONSIDERAÇÕES FINAIS.....</b>	<b>42</b>
6.	<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>43</b>

## 1. INTRODUÇÃO

A ideia deste trabalho acadêmico surgiu em convergir duas grandezas do mundo moderno, de um lado os sistemas operacionais e de outro as memórias de armazenamento.

A problematização deste trabalho acadêmico imerso no universo tecnológico da comunicação, gerou à pergunta: Como ter em mãos, há qualquer momento, em qualquer lugar, sistemas operacionais rodando nos mais diversos equipamentos, desempenhando várias atividades de operações, de maneira segura, econômica e eficiente, e o melhor de modo portátil?

Conseguimos imaginar por exemplo: Um pen drive sem um sistema operacional pronto para ser utilizado, o qual pode rodar a qualquer momento, em qualquer lugar, nos mais variados equipamentos? Pouco sabemos quando vamos necessitar de tal acesso, porém, tê-lo em mãos e em qualquer lugar, e ainda com a opção de utilizar, testar e ou instalar o sistema de forma segura, econômica e eficiente, o torna indispensável.

Realizamos estudos, testes e pesquisas envolvendo soluções completas de sistemas operacionais armazenados em memórias portáteis.

Neste trabalho acadêmico foi detalhado um teste prático em forma de passo-a-passo, envolvendo um sistema operacional livre armazenado em um pen drive, bem como pesquisa qualitativa com o objetivo de localizar e explorar usuários que utilizaram, utilizam e ou virão a utilizar tais soluções, buscando indivíduos que possuem um pen drive ou algum outro dispositivo de armazenamento envolvendo memória sólida portátil não volátil e quais destes indivíduos possuem no mesmo um sistema operacional pronto para ser utilizado. Entrevistas e questionários foram aplicados em indivíduos das mais variadas áreas, as quais estão conectadas de maneira direta e ou indireta a área de tecnologia da informação e comunicação (TIC).

Com esta proposta obtemos atributos específicos neste segmento e compartilhamos informações que colaboram com o desenvolvimento e aprimoram as práticas do uso de sistemas operacionais armazenados em memórias sólidas portáteis não-voláteis, otimizamos técnicas operacionais, produzimos benefícios econômicos, sociais e de segurança com eficiência.

Com os resultados obtidos neste trabalho acadêmico, somado ao avanço das tecnologias e da comunicação, esperamos atrair profissionais e interessados para este tema o qual ainda dispomos de muito campo para explorá-lo, melhoramos as práticas de gestão para um melhor dia-a-dia das pessoas envolvidas com esses processos de uma maneira geral. Acabamos por torná-lo uma referência no âmbito acadêmico e ainda como resultado final mensuramos uma possível base para patente na área.

## **2. PROBLEMATIZAÇÃO**

Como ter em mãos, há qualquer momento, em qualquer lugar, sistemas operacionais rodando nos mais diversos equipamentos, desempenhando várias atividades de operações, de maneira segura, econômica e eficiente, para qualquer tipo de público e o melhor de modo portátil?

### **2.1 OBJETIVOS**

#### **2.1.1 OBJETIVOS GERAIS**

Convergir duas grandezas do mundo moderno, de um lado os sistemas operacionais e de outro as memórias de armazenamento, ambos de uma maneira portátil, prontos para serem utilizados nos mais variados equipamentos.

Conseguimos imaginar por exemplo: Um pen drive sem um sistema operacional pronto para ser utilizado, o qual pode rodar a qualquer momento, em qualquer lugar, nos mais variados equipamentos? Pouco sabemos quando vamos necessitar de tal acesso, porém, tê-lo em mãos e em qualquer lugar, e ainda com a opção de utilizar, testar e ou instalar o sistema de forma segura, econômica e eficiente, o torna indispensável.

#### **2.1.2 OBJETIVOS ESPECÍFICOS**

Buscamos indivíduos que possuem um pen drive ou algum outro dispositivo de armazenamento envolvendo memória sólida portátil não volátil e quais destes

indivíduos possuem no mesmo, um sistema operacional pronto para ser utilizado.

Aplicamos entrevistas e questionários em indivíduos das mais variadas áreas, as quais estão conectadas de maneira direta e ou indireta a área de tecnologia da informação e comunicação (TIC).

Disponibilizamos um tutorial passo-a-passo aos leitores permitindo convergir um sistema operacional com uma memória sólida portátil não-volátil.

## **2.2 DELIMITAÇÕES**

Foram realizados estudos, testes e pesquisas envolvendo soluções completas de sistemas operacionais armazenados em memórias sólidas portáteis não-voláteis, porém, este trabalho acadêmico ficou limitado aos detalhes de um teste envolvendo apenas um, sistema operacional livre armazenado em apenas um, pen drive.

Já a pesquisa empírica de cunho qualitativa que teve como objetivo localizar e explorar usuários que utilizaram, utilizam e ou virão a utilizar tais soluções, também ficou limitada no âmbito da UTFPR - Campus Curitiba apenas aos duzentos indivíduos os quais colaboraram.

## **2.3 METODOLOGIA**

Dividimos o desenvolvimento este trabalho acadêmico em 3 capítulos:

O Capítulo I (tutorial passo-a-passo), realizamos um teste prático em forma de tutorial passo-a-passo, descrito de maneira detalhada, envolvendo um sistema operacional completo e livre armazenado em uma memória sólida portátil não-volátil (Pen drive), sendo uma das possíveis formas de alcançarmos os objetivos deste trabalho. Também descrito os locais onde obter gratuitamente sistemas operacionais e demais informações e assuntos relacionados ao campo da pesquisa.

Já o Capítulo II (pesquisa empírica), expomos alguns dados em forma de tabelas e gráficos, os quais foram obtidos através de uma pesquisa de cunho qualitativa, captando o problema em toda a sua extensão, buscando melhores práticas de gestão e obtendo benefícios práticos. Para sua realização, contamos com a colaboração de duzentos indivíduos que utilizaram, utilizam e ou virão a

utilizar alguma memória sólida portátil não-volátil em seu dia-a-dia, bem como, quais destes indivíduos possuem na mesma um sistema operacional pronto para ser utilizado. Entrevistas e questionários foram aplicados dentro da UTFPR - Campus Curitiba, em duzentos indivíduos das mais variadas áreas, as quais estão conectadas de maneira direta e ou indireta a área de tecnologia da informação e comunicação (TIC).

Por fim, porém, não menos importante, o Capítulo III (fundamentação teórica), no sentido de revisar as bibliografias atuais disponíveis sobre os temas abordados.

### **3. DESENVOLVIMENTO**

#### **3.1 *CAPÍTULO I (TUTORIAL PASSO-A-PASSO)***

Descrição de um teste prático em forma de tutorial passo-a-passo, envolvendo um sistema operacional completo e livre armazenado em uma memória sólida portátil não-volátil (Pen drive). Descrição de locais onde obter gratuitamente sistemas operacionais e demais informações e assuntos relacionados ao campo da pesquisa.

#### **Software Livre:**

A comunidade do software livre no Brasil cresceu de uma maneira assustadora, bem como o compartilhamento de informações pela internet, troca de experiências e vivências na área. Podemos obter mais informações em:

<http://softwarelivre.org/>

"Software livre" se refere à liberdade dos usuários executarem, copiarem, distribuírem, estudarem, modificarem e aperfeiçoarem o software. Mais precisamente, ele se refere a quatro liberdades, para os usuários do software:

- A liberdade de executar o programa, para qualquer propósito (liberdade no. 0);
- A liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades (liberdade no. 1). Acesso ao código-fonte é um pré-requisito para esta liberdade;
- A liberdade de redistribuir cópias de modo que você possa ajudar ao seu próximo (liberdade no. 2);

- A liberdade de aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie (liberdade no. 3). Acesso ao código-fonte é um pré-requisito para esta liberdade.

Fonte: <http://softwarelivre.org/porta/o-que-e>

#### **Notebook utilizado:**

O notebook utilizado no teste é da marca Sony, modelo Vaio, E-350 VPCYB25AB, com o sistema operacional proprietário da empresa Microsoft - Windows 7 Starter - Product Key – 2PDC9 – F3JM7 – QCCG7 – VPGVJ – 9968V, apesar do sistema operacional trabalhar apenas com 2GB de memória RAM, foi acrescentado mais um pente em seu hardware de 2GB de memória RAM, totalizando 4GB de memória RAM, e o seu HD que vem de fábrica foi substituído por um SSD da marca OCZ - Modelo - Vertex 4.

#### **Memória sólida portátil não-volátil utilizada:**

Necessitamos de um pen drive para realizar o processo prático descrito neste trabalho acadêmico. Foi utilizado um pen drive da marca Kingston, à Kingston Technology Co. é uma empresa norte-americana fabricante de componentes para computadores, principalmente de memórias para computador, mais informações em: <http://www.kingston.com/br/> o modelo utilizado foi o DT101 G2, com capacidade de 16GB, porém, após sua formatação para o sistema de arquivos FAT32, ficou disponível no total de espaço 15.461.941.248 bytes, foram necessários apenas 1GB, mais precisamente 1.045.225.472 bytes do total de espaço disponível da memória para acomodar os dados necessários e deixá-lo preparado para inicializar o sistema operacional quando lhe for oportuno. Ainda restou livres 14GB, mais precisamente 14.416.715.776 bytes para utilizar com outros dados. Também, realizamos testes utilizando uma memória microSD da marca Lexar, modelo com capacidade de 2GB, ligada ao notebook através de um adaptador microSD para USB, onde foi utilizada com sucesso, porém, não deixamos mais detalhes neste tutorial.

#### **Sistema operacional utilizado:**

Foram feitos testes com vários sistemas, porém, o escolhido para este passo-a-passo foi o Ubuntu, na versão 14.04.2 Trusty Tahr LTS, totalmente estável, disponível no endereço eletrônico: <http://www.ubuntu.com/download/desktop> -

Acessado em 25/02/2015, com esforços realizados pela Canonical Ltd. Uma empresa fundada pelo sul-africano Mark Shuttleworth em 5 de março de 2004, sediada na Ilha de Man, Reino Unido, a qual trabalha na promoção do software livre.

Para mais informações, o Release Notes está disponível no endereço eletrônico: <https://wiki.ubuntu.com/TrustyTahr/ReleaseNotes>

Ou ainda, para mais informações do sistema operacional utilizado neste tutorial acesse: <http://distrowatch.com/?newsid=08820>

Escolhido este pelo simples fato de ser um dos mais difundidos e utilizados para diversos fins, a comunidade do software livre, vem por muitos anos realizando esforços para torná-lo cada vez menos incompatível com os hardwares atuais, o mesmo vem acompanhado de vários softwares pré-instalados e pré-configurados para serem utilizados de um modo que permita uma experiência ímpar, ou seja, testar, instalar e utilizar de maneira gratuita.

#### **Mais Informações sobre sistemas operacionais livres:**

Deixamos aqui o endereço na internet para o site: <http://distrowatch.com/> onde dispõe de um banco de dados significativo com muitas informações completas referente as mais variadas distribuições de sistemas operacionais livres disponíveis no mundo.

#### **Software utilizado, Universal USB Installer:**

Informações do software utilizado neste trabalho acadêmico, o qual permite transferir a imagem do sistema operacional para o pen drive, detalhes de como utilizar e como obtê-lo gratuitamente, podem ser obtidas no endereço eletrônico: <http://www.pendrivelinux.com/universal-usb-installer-easy-as-1-2-3/>

Existem vários softwares disponíveis para transformar as imagens baixadas dos sistemas em arquivos de inicialização e incluí-las nas memórias sólidas portáteis não-voláteis, alguns gratuitos como o utilizado neste tutorial como também softwares proprietários. Ex: UltraISO, WinToUSB, Fedora Live USB Creator, UNetbootin, entre outros. Para uma lista mais completa acesse: [http://en.wikipedia.org/wiki/List\\_of\\_tools\\_to\\_create\\_Live\\_USB\\_systems](http://en.wikipedia.org/wiki/List_of_tools_to_create_Live_USB_systems)

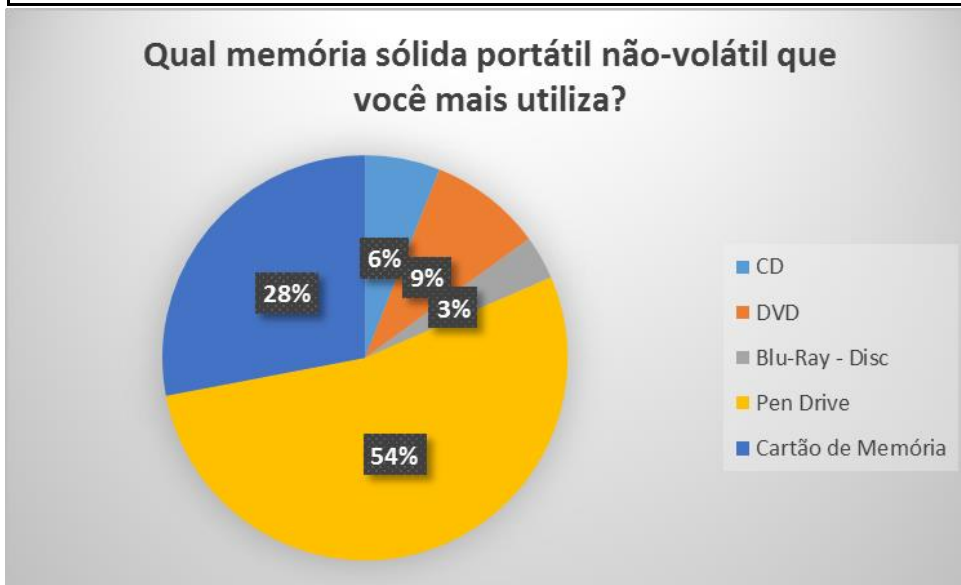


### 3.2 CAPÍTULO II (PESQUISA EMPÍRICA)

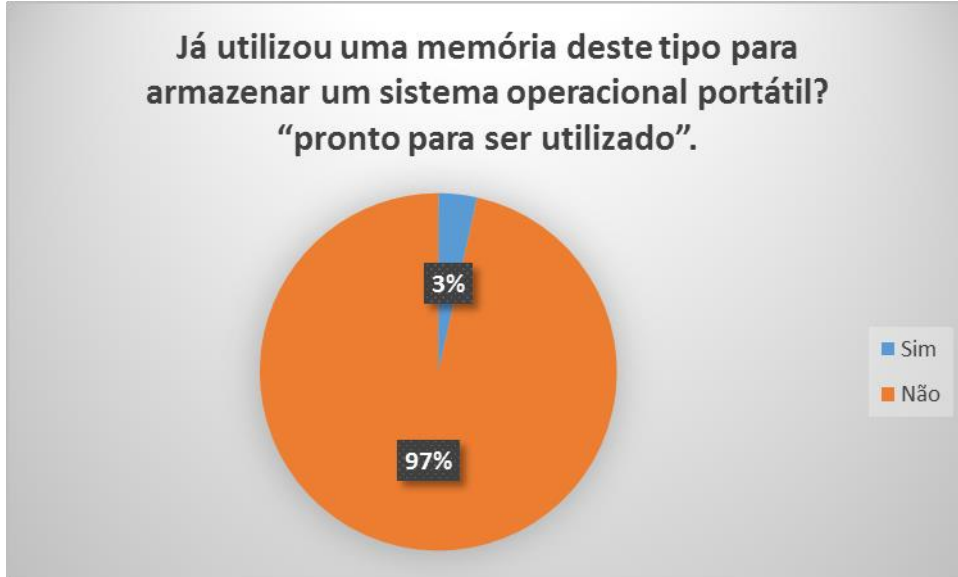
Pesquisa de cunho qualitativa buscando indivíduos que utilizam alguma memória sólida portátil não-volátil no dia-a-dia e quais destes indivíduos possuem no mesmo um sistema operacional pronto para ser utilizado.

Perguntas efetuadas aos indivíduos:

Qual memória sólida portátil não-volátil você mais utiliza?	Resposta
CD	12
DVD	18
Blu-Ray – Disc	7
Pen Drive	107
Cartão de Memória	56

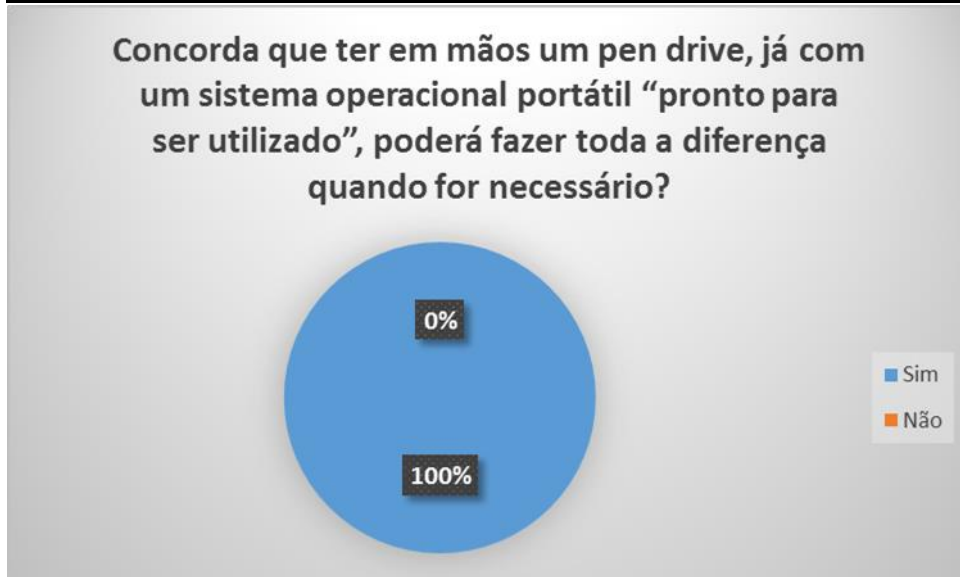


Já utilizou uma memória deste tipo para armazenar um sistema operacional portátil? "Pronto para ser utilizado".	Resposta
Sim	7
Não	193



*OBS: Todas as respostas "sim" para a pergunta utilizaram como memória o CD.  
Fonte: o autor*

Concorda que ter em mãos um pen drive, já com um sistema operacional portátil “pronto para ser utilizado”, poderá fazer toda a diferença quando for necessário?	Resposta
Sim	200
Não	0



### 3.3 CAPÍTULO III - (FUNDAMENTAÇÃO TEÓRICA)

Convergir um sistema operacional à uma memória sólida portátil não-volátil é uma tarefa um tanto quanto complexa, porém, segundo [SILBERSCHATZ, 2012], nunca houve uma época tão interessante para estudar os sistemas operacionais e nunca foi tão fácil. O movimento do código-fonte aberto tomou conta dos sistemas operacionais, fazendo com que muitos deles fossem disponibilizados tanto no formato de código-fonte quanto no formato binário (executável).

A memória Flash se tornou uma das tecnologias mais importantes das últimas décadas, possibilitando o surgimento dos cartões de memória, pendrives, SSDs, celulares, câmeras, e players de mídia com armazenamento interno e assim por diante. Se a memória Flash não existisse, todas essas áreas estariam muito atrasadas em relação ao que temos hoje. [MORIMOTO, 2013].

Um sistema de computação é constituído basicamente por hardware e software. O hardware é composto por circuitos eletrônicos (processador, memória, portas de entrada/saída, etc.) e periféricos eletro-óptico-mecânicos (teclados, mouses, discos rígidos, unidades de disquete, CD ou DVD, dispositivos USB, etc.). Por sua vez, o software de aplicação é representado por programas destinados ao usuário do sistema, que constituem a razão final de seu uso, como editores de texto, navegadores Internet ou jogos. Entre os aplicativos e o hardware reside uma camada de software multi-facetada e complexa, denominada genericamente de Sistema Operacional. [MAZIERO, 2013].

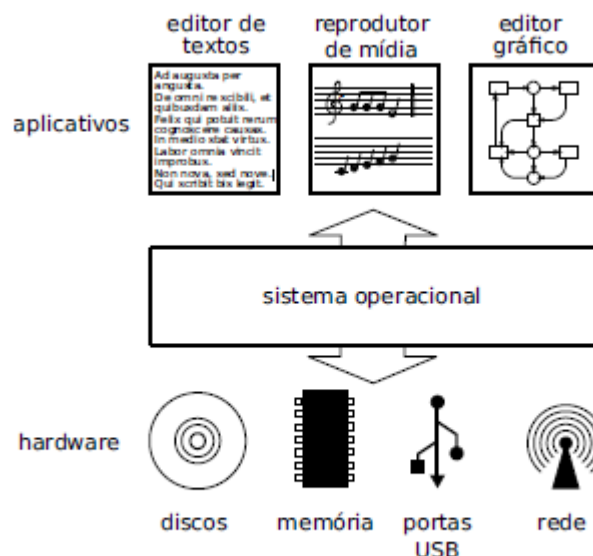


Figura 1: Estrutura de um sistema de computação típico

Fonte: MAZIERO, 2013, (*Sistemas Operacionais: Conceitos e Mecanismos*, p. 4)

### 3.3.1 HISTÓRIA DOS SISTEMAS OPERACIONAIS

Os primeiros sistemas de computação, no final dos anos 40 e início dos anos 50, não possuíam sistema operacional. Por outro lado, os sistemas de computação atuais possuem sistemas operacionais grandes, complexos e em constante evolução. A seguir são apresentados alguns dos marcos mais relevantes na história dos sistemas operacionais [*Foundation, 2005*]:

**Anos 40:** cada programa executava sozinho e tinha total controle do computador. A carga do programa em memória, a varredura dos periféricos de entrada para busca de dados, a computação propriamente dita e o envio dos resultados para os periféricos de saída, byte a byte, tudo devia ser programado detalhadamente pelo desenvolvedor da aplicação.

**Anos 50:** os sistemas de computação fornecem “bibliotecas de sistema” (system libraries) que encapsulam o acesso aos periféricos, para facilitar a programação de aplicações. Algumas vezes um programa “monitor” (system monitor) auxilia a carga e descarga de aplicações e/ou dados entre a memória e periféricos (geralmente leitoras de cartão perfurado, fitas magnéticas e impressoras de caracteres).

**1961:** o grupo do pesquisador Fernando Corbató, do MIT, anuncia o desenvolvimento do CTSS – Compatible Time-Sharing System [Corbató et al., 1962], o primeiro sistema operacional com compartilhamento de tempo.

**1965:** a IBM lança o OS/360, um sistema operacional avançado, com compartilhamento de tempo e excelente suporte a discos. 26 c Carlos Maziero: Um breve histórico dos sistemas operacionais.

**1965:** um projeto conjunto entre MIT, GE e Bell Labs define o sistema operacional Multics, cujas ideias inovadoras irão influenciar novos sistemas durante décadas.

**1969:** Ken Thompson e Dennis Ritchie, pesquisadores dos Bell Labs, criam a primeira versão do UNIX.

**1981:** a Microsoft lança o MS-DOS, um sistema operacional comprado da

empresa Seattle Computer Products em 1980.

**1984:** a Apple lança o sistema operacional Macintosh OS 1.0, o primeiro a ter uma interface gráfica totalmente incorporada ao sistema.

**1985:** primeira tentativa da Microsoft no campo dos sistemas operacionais com interface gráfica, através do MS-Windows 1.0.

**1987:** Andrew Tanenbaum, um professor de computação holandês, desenvolve um sistema operacional didático simplificado, mas respeitando a API do UNIX, que foi batizado como Minix.

**1987:** IBM e Microsoft apresentam a primeira versão do OS/2, um sistema multitarefa destinado a substituir o MS-DOS e o Windows. Mais tarde, as duas empresas rompem a parceria; a IBM continua no OS/2 e a Microsoft investe no ambiente Windows.

**1991:** Linus Torvalds, um estudante de graduação finlandês, inicia o desenvolvimento do Linux, lançando na rede Usenet o núcleo 0.01, logo abraçado por centenas de programadores ao redor do mundo.

**1993:** a Microsoft lança o Windows NT, o primeiro sistema 32 bits da empresa.

**1993:** lançamento dos UNIX de código aberto FreeBSD e NetBSD.

**2001:** a Apple lança o MacOS X, um sistema operacional derivado da família UNIX BSD.

**2001:** lançamento do Windows XP.

**2004:** lançamento do núcleo Linux 2.6.

**2006:** lançamento do Windows Vista.

Esse histórico reflete apenas o surgimento de alguns sistemas operacionais relativamente populares; diversos sistemas acadêmicos ou industriais de grande importância pelas contribuições inovadoras, como Mach, Chorus, QNX e Plan 9, não estão representados. [MAZIERO, 2013].

### 3.3.2 SISTEMA OPERACIONAL

Segundo [TANENBAUM, 2012], sem software, um computador é basicamente um monte inútil de metal. Com software, um computador pode armazenar, processar e recuperar informações, tocar música e reproduzir vídeos, enviar e-mail, pesquisar a Internet e se envolver em muitas outras

atividades valiosas para merecer sua manutenção. Grosso modo, o software de computador pode ser dividido em dois tipos: programas de sistema, que gerenciam a operação do computador em si, e programas aplicativos, que realizam o trabalho real desejado pelo usuário. O programa de sistema mais básico é o sistema operacional, cuja tarefa é controlar todos os recursos do computador e fornece uma base sobre a qual os programas aplicativos podem ser escritos.

Um sistema de computação moderno consiste em um ou mais processadores, memória principal, discos, impressoras, teclado, tela, interfaces de rede e outros dispositivos de entrada/saída. No todo, um sistema complexo. Escrever programas que controlam todos esses componentes e os utilizam corretamente, sem dizer de forma otimizada, é uma tarefa extremamente difícil. Se todo programador tivesse que se preocupar com o funcionamento das unidades de discos e com todas as dezenas de coisas que poderiam dar errado ao ler um bloco de disco, é provável que muitos programas sequer pudessem ser escritos. Há muito tempo, tornou-se bastante evidente a necessidade de encontrar uma maneira de isolar os programadores da complexidade do hardware. A maneira que evoluiu gradualmente foi colocar uma camada de software sobre o do hardware básico, para gerenciar todas as partes do sistema e apresentar ao usuário uma interface, ou máquina virtual, mais fácil de entender e programar.

Essa camada de software é o sistema operacional.

*Fonte: (TANENBAUM e WOODHULL – Sistemas operacionais – Projeto e Implementação – terceira edição - 2012).*

Para [SILBERSCHATZ, 2012], um sistema operacional é um programa que gerencia o hardware do computador. Ele também fornece uma base para os programas aplicativos e atua como intermediário entre o usuário e o hardware do computador. Um aspecto interessante dos sistemas operacionais é o quanto eles assumem diferentes abordagens ao cumprir essas tarefas. Os sistemas operacionais de mainframe são projetados basicamente para otimizar a utilização do hardware. Os sistemas operacionais dos computadores pessoais (PCs) suportam jogos complexos, aplicações comerciais e tudo o mais entre eles. Os sistemas operacionais de computadores móveis são projetados de modo a oferecer um ambiente no qual o usuário possa interagir facilmente com o computador para executar programas. Assim, alguns sistemas operacionais são projetados para ser convenientes, outros para ser eficientes, e outros para atender a alguma combinação de ambos os aspectos.

Internamente, os sistemas operacionais variam muito em sua composição, já que estão organizados em muitas linhas diferentes. O projeto de um novo sistema operacional é uma tarefa de peso. É importante que os objetivos do sistema sejam bem definidos antes de o projeto começar. Esses objetivos formam a base das escolhas feitas entre vários algoritmos e estratégias. Já que um sistema operacional é grande e complexo, deve ser criado por módulos. Cada um desses módulos deve ser uma parcela delineada do sistema, com entradas, saídas e funções definidas cuidadosamente.

*Fonte: (Fundamentos de Sistemas Operacionais – SILBERSCHATZ, Abraham; GALVIN, Peter Baer; GAGNE, Greg) – 8ª Edição, 2012.*

Conforme, [MACHADO e MAIA, 2013], um sistema operacional, por mais complexo que possa parecer, é apenas um conjunto de rotinas executado pelo processador, de forma semelhante aos programas dos usuários. Sua principal função é controlar o funcionamento de um computador, gerenciando a utilização e o compartilhamento dos seus diversos recursos, como processadores, memórias e dispositivos de entrada e saída.

Sem o sistema operacional, um usuário para interagir com o computador deveria conhecer profundamente diversos detalhes sobre hardware do equipamento, o que tornaria seu trabalho lento e com grandes possibilidades de erros. O sistema operacional tem como objetivo funcionar como uma interface entre o usuário e o computador, tornando sua utilização mais simples, rápida e segura.

*Fonte: (MACHADO, Francis Berenger; MAIA, Luiz Paulo - Arquitetura de sistemas operacionais). 5ª edição. 2013.*

A história segundo [TANENBAUM, 2012], tem a tendência de se repetir no mundo da computação: inicialmente, o software de um minicomputador era como o software de um computador de grande porte e, posteriormente, o software de um computador pessoal era como o software de um minicomputador. Agora o ciclo está se repetindo com os palmtops, PDAs e sistemas embarcados. Nesses sistemas, ainda estão em uso esquemas de gerenciamento de memórias simples. Por isso, ainda vale a pena estudá-los.

Sistemas mais simples não fazem swapping nem paginação. Uma vez que um programa é carregado na memória, ele permanece lá até terminar. Os sistemas embarcados normalmente funcionam assim, possivelmente até com o código na memória ROM. Alguns sistemas operacionais permitem apenas um processo por vez na memória, enquanto outros suportam multiprogramação.



### 3.3.3 TIPOS DE SISTEMAS OPERACIONAIS

MAZIERO, (2013), descreve os tipos de sistemas operacionais como: Os sistemas operacionais podem ser classificados segundo diversos parâmetros e perspectivas, como tamanho, velocidade, suporte a recursos específicos, acesso à rede, etc. A seguir são apresentados alguns tipos de sistemas operacionais usuais (muitos sistemas operacionais se encaixam bem em mais de uma das categorias apresentadas):

**Batch (de lote):** os sistemas operacionais mais antigos trabalhavam “por lote”, ou seja, todos os programas a executar eram colocados em uma fila, com seus dados e demais informações para a execução. O processador recebia os programas e os processava sem interagir com os usuários, o que permitia um alto grau de utilização do sistema. Atualmente, este conceito se aplica a sistemas que processam tarefas sem interação direta com os usuários, como os sistemas de processamento de transações em bancos de dados. Além disso, o termo “em lote” também é usado para designar um conjunto de comandos que deve ser executado em sequência, sem interferência do usuário. Exemplos desses sistemas incluem o OS/360 e VMS, entre outros.

**De rede:** um sistema operacional de rede deve possuir suporte à operação em rede, ou seja, a capacidade de oferecer às aplicações locais recursos que estejam localizados em outros computadores da rede, como arquivos e impressoras. Ele também deve disponibilizar seus recursos locais aos demais computadores, de forma controlada. A maioria dos sistemas operacionais atuais oferece esse tipo de funcionalidade.

**Distribuído:** em um sistema operacional distribuído, os recursos de cada máquina estão disponíveis globalmente, de forma transparente aos usuários. Ao lançar uma aplicação, o usuário interage com sua janela, mas não sabe onde ela está executando ou armazenando seus arquivos: o sistema é quem decide, de forma transparente. Os sistemas operacionais distribuídos já existem há tempos (Amoeba [Tanenbaum et al., 1991] e Clouds [Dasgupta et al., 1991],

por exemplo), mas ainda não são uma realidade de mercado.

**Multi-usuário:** um sistema operacional multi-usuário deve suportar a identificação do “dono” de cada recurso dentro do sistema (arquivos, processos, áreas de memória, conexões de rede) e impor regras de controle de acesso para impedir o uso desses recursos por usuários não autorizados. Essa funcionalidade é fundamental para a segurança dos sistemas operacionais de rede e distribuídos. Grande parte dos sistemas atuais são multi-usuários.

**Desktop:** um sistema operacional “de mesa” é voltado ao atendimento do usuário doméstico e corporativo para a realização de atividades corriqueiras, como edição de textos e gráficos, navegação na Internet e reprodução de mídias simples. Suas principais características são a interface gráfica, o suporte à interatividade e a operação em rede. Exemplos de sistemas desktop são os vários sistemas Windows (XP, Vista, 7, etc.), o MacOS X e Linux.

**Servidor:** um sistema operacional servidor deve permitir a gestão eficiente e de grandes quantidades de recursos (disco, memória, processadores), impondo prioridades e limites sobre o uso dos recursos pelos usuários e seus aplicativos. Normalmente um sistema operacional servidor também tem suporte a rede e multi-usuários.

**Embarcado:** um sistema operacional é dito embarcado (embutido ou embedded) quando é construído para operar sobre um hardware com poucos recursos de processamento, armazenamento e energia. Aplicações típicas desse tipo de sistema aparecem em telefones celulares, sistemas de automação industrial e controladores automotivos, equipamentos eletrônicos de uso doméstico (leitores de DVD, TVs, fornos-micro-ondas, centrais de alarme, etc.). Muitas vezes um sistema operacional embarcado se apresenta na forma de uma biblioteca a ser ligada ao programa da aplicação (que é fixa). LynxOS, \_C/OS, Xylinx e VxWorks são exemplos de sistemas operacionais embarcados para controle e automação. Sistemas operacionais para telefones celulares inteligentes (smartphones) incluem o Symbian e o Android, entre outros.

**Tempo real:** ao contrário da concepção usual, um sistema operacional de tempo real não precisa ser necessariamente ultra-rápido; sua característica essencial é ter um comportamento temporal previsível (ou seja, seu tempo de resposta deve ser conhecido no melhor e pior caso de operação). A estrutura interna de um sistema operacional de tempo real deve ser construída de forma a minimizar esperas e latências imprevisíveis, como tempos de acesso a disco e sincronizações excessivas.

Existem duas classificações de sistemas de tempo real: soft real-time systems, nos quais a perda de prazos implica na degradação do serviço prestado. Um exemplo seria o suporte à gravação de CDs ou à reprodução de músicas. Caso o sistema se atrase, pode ocorrer a perda da mídia em gravação ou falhas na música que está sendo tocada. Por outro lado, nos hard real-time systems a perda de prazos pelo sistema pode perturbar o objeto controlado, com graves consequências humanas, econômicas ou ambientais. Exemplos desse tipo de sistema seriam o controle de funcionamento de uma turbina de avião a jato ou de uma caldeira industrial.

Exemplos de sistemas de tempo real incluem o QNX, RT-Linux e VxWorks. Muitos sistemas embarcados têm características de tempo real, e vice-versa. [Maziero, 2013].

Já para [MACHADO e MAIA, 2013], os tipos de sistemas operacionais e sua evolução estão relacionados diretamente com a evolução do hardware e das aplicações por ele suportadas. Muitos termos inicialmente introduzidos para definir conceitos e técnicas foram substituídos por outros, na tentativa de refletir uma nova maneira de interação ou processamento. Isto fica muito claro quando tratamos da unidade de execução do processador. Inicialmente, os termos programa ou job eram os mais utilizados, depois surgiu o conceito de processo e subprocesso e, posteriormente, o conceito de thread.

### **Sistemas Monoprogramáveis/Monotarefa**

Os primeiros sistemas operacionais eram tipicamente voltados para a execução de um único programa. Qualquer outra aplicação, para ser

executada, deveria aguardar o término do programa corrente. Os sistemas monoprogramáveis, como vieram a ser conhecidos, se caracterizam por permitir que o processador, a memória e os periféricos permaneçam exclusivamente dedicados à execução de um único programa.

Os sistemas monoprogramáveis estão tipicamente relacionados ao surgimento dos primeiros computadores na década de 1960. Posteriormente, com a introdução dos computadores pessoais e estações de trabalho na década de 1970, este tipo de sistema voltou a ser utilizado para atender máquinas que, na época, eram utilizadas por apenas um usuário. Os sistemas monotarefa, como também são chamados, caracterizam-se por permitir que todos os recursos do sistema fiquem exclusivamente dedicados a uma única tarefa.

Neste tipo de sistema, enquanto um programa aguarda por um evento, como a digitação de um dado, o processador permanece ocioso, sem realizar qualquer tipo de processamento. A memória é subutilizada caso o programa não a preencha totalmente, e os periféricos, como discos e impressoras, estão dedicados a um único usuário, nem sempre utilizados de forma integral.

Comparado a outros sistemas, os sistemas monoprogramáveis ou monotarefa são de simples implementação, não existindo muita preocupação com problemas decorrentes do compartilhamento de recursos, como memória, processador e dispositivos de E/S.

### **Sistemas Multiprogramáveis/Multitarefa**

Os sistemas multiprogramáveis ou multitarefas são uma evolução dos sistemas monoprogramáveis. Neste tipo de sistema, os recursos computacionais são compartilhados entre os diversos usuários e aplicações. Enquanto em sistemas monoprogramáveis existe apenas um programa utilizando os recursos disponíveis, nos multiprogramáveis várias aplicações compartilham esses mesmos recursos.

Neste tipo de sistema, por exemplo, enquanto um programa espera por uma operação de leitura ou gravação em disco, outros programas podem estar sendo processados neste mesmo intervalo de tempo. Nesse caso, podemos

observar o compartilhamento da memória e do processador. O sistema operacional se preocupa em gerenciar o acesso concorrente aos seus diversos recursos, como memória, processador e periféricos, de forma ordenada e protegida, entre os diversos programas.

A principal vantagem dos sistemas multiprogramáveis é a redução de custos em função da possibilidade do compartilhamento dos diversos recursos entre as diferentes aplicações. Além disso, sistemas multiprogramáveis, apesar de mais eficientes que os monoprogramáveis, são de implementação muito mais complexa.

A partir do número de usuários que interagem com o sistema operacional, podemos classificar os sistemas multiprogramáveis como monousuário ou multiusuário. Sistemas multiprogramáveis monousuário são encontrados em computadores pessoais e estações de trabalho, onde há apenas um único usuário interagindo com o sistema. Neste caso existe a possibilidade da execução de diversas tarefas ao mesmo tempo, como a edição de um texto, uma impressão e o acesso à Internet. Sistemas multiprogramáveis multiusuário são ambientes interativos que possibilitam a diversos usuários conectarem-se ao sistema simultaneamente.

Os sistemas multiprogramáveis ou multitarefa podem ser classificados pela forma com que suas aplicações são gerenciadas, podendo ser divididos em sistemas batch, de tempo compartilhado ou de tempo real. Um sistema operacional pode suportar um ou mais desses tipos de processamento, dependendo de sua implementação.

### **Sistemas batch**

Os sistemas batch foram os primeiros tipos de sistemas operacionais multiprogramáveis a serem implementados da década de 1960. Os programas, também chamados de jobs, eram submetidos para execução através de cartões perfurados e armazenados em disco ou fita, onde aguardavam para ser processados. Posteriormente, em função da disponibilidade de espaço na memória principal, os jobs eram executados, produzindo uma saída em disco ou fita.

O processamento batch tem a característica de não exigir a interação do usuário com a aplicação. Todas as entradas e saídas de dados da aplicação são implementadas por algum tipo de memória secundária, geralmente arquivos em disco. Alguns exemplos de aplicações originalmente processadas em batch são programas envolvendo cálculos numéricos, compilações, ordenações, backups e todos aqueles onde não é necessária a interação com o usuário.

Esses sistemas, quando bem projetados, podem ser bastante eficientes, devido à melhor utilização do processador, entretanto podem oferecer tempos de resposta longos. Atualmente, os sistemas operacionais implementam ou simulam o processamento batch, não existindo sistemas exclusivamente dedicados a este tipo de processamento.

### **Sistemas de tempo compartilhado**

Os sistemas de tempo compartilhado (time-sharing) permitem que diversos programas sejam executados a partir da divisão do tempo do processador em pequenos intervalos, denominada fatia de tempo (time-slice). Caso a fatia de tempo não seja suficiente para a conclusão do programa, ele é interrompido pelo sistema operacional e substituído por um outro, enquanto fica aguardando por uma nova fatia de tempo. O sistema cria para cada usuário um ambiente de trabalho próprio dando a impressão de que todo o sistema está dedicado exclusivamente a ele.

Geralmente, sistemas de tempo compartilhado permitem a interação dos usuários com o sistema através de terminais que incluem vídeo, teclado e mouse. Esses sistemas possuem uma linguagem de controle que permite ao usuário comunicar-se diretamente como o sistema operacional através de comandos. Desta forma é possível verificar arquivos armazenados em disco ou cancelar a execução de um programa. O sistema, normalmente, responde em poucos segundos à maioria desses comandos. Devido a esse tipo de interação, os sistemas de tempo compartilhado também ficaram conhecidos como sistemas on-line.

A maioria das aplicações comerciais atualmente é processada em sistemas de tempo compartilhado, pois elas oferecem tempos de respostas

razoáveis a seus usuários e custos mais baixos em função da utilização compartilhada dos diversos recursos do sistema.

### **Sistemas de tempo real**

Ainda segundo, [MACHADO e MAIA, 2013], os sistemas de tempo real (real-time) são implementados de forma semelhante aos sistemas de tempo compartilhado. O que caracteriza a diferença entre os dois tipos de sistemas é o tempo exigido no processamento das aplicações. Enquanto em sistemas de tempo compartilhado o tempo de processamento pode variar sem comprometer as aplicações em execução, nos sistemas de tempo real os tempos de processamento devem estar dentro de limites rígidos, que devem ser obedecidos, caso contrário poderão ocorrer problemas irreparáveis.

Nos sistemas de tempo real não existe a ideia de fatia de tempo, implementada nos sistemas de tempo compartilhado. Um programa utiliza o processador o tempo que for necessário ou até que apareça outro mais prioritário. A importância ou prioridade de execução de um programa é definida pela própria aplicação e não pelo sistema operacional.

Esses sistemas, normalmente, estão presentes em aplicações de controle de processos, como no monitoramento de refinarias de petróleo, controle de tráfego aéreo, de usinas termoelétricas e nucleares, ou em qualquer aplicação em que o tempo de processamento é fator fundamental.

### **Sistemas com Múltiplos Processadores**

Os sistemas com múltiplos processadores caracterizam-se por possuir duas ou mais UCPs interligadas e trabalhando em conjunto. A vantagem deste tipo de sistema é permitir que vários programas sejam executados ao mesmo tempo ou que um mesmo programa seja subdividido em partes para serem executadas simultaneamente em mais de um processador.

Com múltiplos processadores foi possível a criação de sistemas computacionais voltados, principalmente, para processamento científico, aplicado, por exemplo, no desenvolvimento aeroespacial, prospecção de

petróleo, simulações, processamento de imagens e CAD. A princípio qualquer aplicação que faça uso intensivo da UCP será beneficiada pelo acréscimo de processadores ao sistema. A evolução desses sistemas deve-se, em grande parte, ao elevado custo de desenvolvimento de processadores de alto desempenho.

Os conceitos aplicados ao projeto de sistemas em múltiplos processadores incorporam os mesmos princípios básicos e benefícios apresentados na multiprogramação, além de outras características e vantagens específicas como escalabilidade, disponibilidade e balanceamento de carga.

MACHADO e MAIA, 2013, explica que a escalabilidade é a capacidade de ampliar o poder computacional do sistema apenas adicionando novos processadores. Em ambientes com um único processador, caso haja problemas de desempenho, seria necessário substituir todo o sistema por uma outra configuração com maior poder de processamento. Com a possibilidade de múltiplos processadores, basta acrescentar novos processadores à configuração.

Disponibilidade é a capacidade de manter o sistema em operação mesmo em casos de falhas. Neste caso, se um dos processadores falhar, os demais podem assumir suas funções de maneira transparente aos usuários e suas aplicações, embora com menor capacidade de computação.

Balanceamento de carga é a possibilidade de distribuir o processamento entre os diversos processadores da configuração a partir da carga de trabalho de cada processador, melhorando, assim, o desempenho do sistema como um todo.

Um fator-chave no desenvolvimento de sistemas operacionais com múltiplos processadores é a forma de comunicação entre as UCPs e o grau de compartilhamento da memória e dos dispositivos de entrada e saída. Em função desses fatores, podemos classificar os sistemas com múltiplos processadores em fortemente acoplados ou francamente acoplados.

A grande diferença entre os dois tipos de sistemas é que em sistemas fortemente acoplados existe apenas uma memória principal sendo compartilhada por todos os processadores, enquanto nos francamente acoplados cada sistema tem sua própria memória individual. Além disso, a taxa de transferência entre processadores e memória em sistemas fortemente



acoplados é muito maior que nos francamente acoplados.

Fonte: (MACHADO, Francis Berenger; MAIA, Luiz Paulo. - *Arquitetura de sistemas operacionais*). 5ª edição. 2013.

### 3.3.3.1 FUNCIONALIDADES

Para cumprir seus objetivos de abstração e gerência, o sistema operacional deve atuar em várias frentes. Cada um dos recursos do sistema possui suas particularidades, o que impõe exigências específicas para gerenciar e abstrair os mesmos. Sob esta perspectiva, as principais funcionalidades implementadas por um sistema operacional típico são:

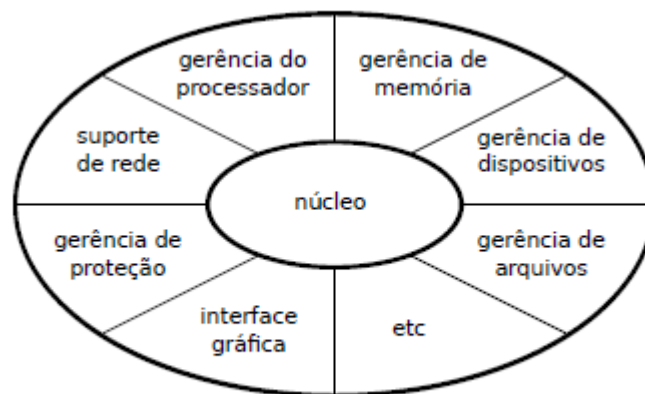


Figura 2: Funcionalidades do sistema operacional

Fonte: MAZIERO, 2013. (*Sistemas Operacionais: Conceitos e Mecanismos*, p. 9)

Distribuir de forma justa, mas não necessariamente igual, pois as aplicações têm demandas de processamento distintas; por exemplo, um navegador de Internet demanda menos do processador que um aplicativo de edição de vídeo, e por isso o navegador pode receber menos tempo de processador. [MAZIERO, 2013].

### 3.3.4 ESTRUTURA DE UM SISTEMA OPERACIONAL

Ainda segundo [MAZIERO, 2013], um sistema operacional não é um bloco único e fechado de software executando sobre o hardware. Na verdade, ele é composto de diversos componentes com objetivos e funcionalidades

complementares. Alguns dos componentes mais relevantes de um sistema operacional típico são:

**Núcleo:** é o coração do sistema operacional, responsável pela gerência dos recursos do hardware usados pelas aplicações. Ele também implementa as principais abstrações utilizadas pelos programas aplicativos.

**Drivers:** módulos de código específicos para acessar os dispositivos físicos. Existe um driver para cada tipo de dispositivo, como discos rígidos IDE, SCSI, portas USB, placas de vídeo, etc. Muitas vezes o driver é construído pelo próprio fabricante do hardware e fornecido em forma compilada (em linguagem de máquina) para ser acoplado ao restante do sistema operacional.

**Código de inicialização:** a inicialização do hardware requer uma série de tarefas complexas, como reconhecer os dispositivos instalados, testá-los e configurá-los adequadamente para seu uso posterior. Outra tarefa importante é carregar o núcleo do sistema operacional em memória e iniciar sua execução.

**Programas utilitários:** são programas que facilitam o uso do sistema computacional, fornecendo funcionalidades complementares ao núcleo, como formatação de discos e mídias, configuração de dispositivos, manipulação de arquivos (mover, copiar, apagar), interpretador de comandos, terminal, interface gráfica, gerência de janelas, etc.

As diversas partes do sistema operacional estão relacionadas entre si conforme apresentado na Figura 3. [Maziero, 2013].

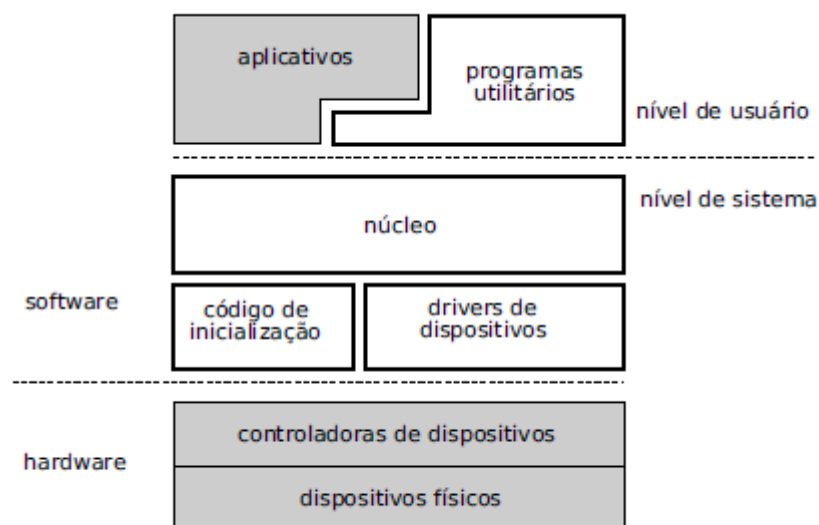


Figura 3: Estrutura de um sistema operacional

Fonte: Maziero (2013, *Sistemas Operacionais: Conceitos e Mecanismos*, p. 11)

### 3.3.5 DEFINIÇÕES E NOTAÇÃO DE ARMAZENAMENTO

SILBERSCHATZ, Abraham; GALVIN, Peter Baer e GAGNE, Greg, 2012, definem que um bit é a unidade básica de armazenamento nos computadores. Ele pode conter um entre dois valores, zero e um. Todos os outros tipos de armazenamento em um computador são baseados em conjuntos de bits. Quando há bits suficientes, é espantoso quantas coisas um computador pode representar: números, letras, imagens, filmes, sons, documentos e programas, para citar apenas algumas. Um byte compõe-se de 8 bits e na maioria dos computadores é o menor bloco de armazenamento conveniente. Por exemplo, a maioria dos computadores não tem uma instrução para mover um bit, e sim para mover um byte. Um termo menos comum é palavra, que é a unidade de armazenamento original de uma determinada arquitetura de computador. Geralmente uma palavra é composta por um ou mais bytes. Por exemplo, um computador pode ter instruções para mover palavras de 64 bits (8 bytes).

Um kilobyte ou KB é igual a  $1.024$  bytes, um megabyte ou MB equivale a  $1.024^2$  bytes, e um gigabyte ou GB é o mesmo que  $1.024^3$  bytes. Os fabricantes de computadores costumam arredondar esses números e dizem que um megabyte corresponde a 1 milhão de bytes em um gigabyte a 1 bilhão de bytes.

### 3.3.6 O ESTUDO DOS SISTEMAS OPERACIONAIS

Nunca houve uma época tão interessante para estudar os sistemas operacionais e nunca foi tão fácil. O movimento do código-fonte aberto tomou conta dos sistemas operacionais, fazendo com que muitos deles fossem disponibilizados tanto no formato de código-fonte quanto no formato binário (executável). Essa lista inclui o Linux, o BSD UNIX, o Solaris e parte do Mac OS X. A disponibilidade do código-fonte nos permite estudar os sistemas operacionais de dentro para fora. Perguntas que antes só podiam ser respondidas através da verificação da documentação ou comportamento de um sistema operacional já podem ser respondidas através da verificação do próprio

código.

Além disso, o surgimento da virtualização como uma função da computação popular (e frequentemente gratuita) torna possível a execução de vários sistemas operacionais no topo de um sistema core. Por exemplo, a VMware (<http://www.vmware.com>) fornece um “player” gratuito em que centenas de “aplicações virtuais” podem ser executadas. Usando esse método, os alunos podem testar centenas de sistemas operacionais de seus próprios sistemas sem custo.

Sistemas operacionais que não são mais comercialmente viáveis também passaram a ter o código-fonte aberto, permitindo-nos estudar como os sistemas operavam em uma época de menos recursos de CPU, memória e espaço de armazenamento. Uma extensa porém incompleta lista de projetos de sistemas operacionais de código-fonte aberto está disponível em [http://dmoz.org/Open\\_Source/](http://dmoz.org/Open_Source/). Simuladores de hardware específico também estão disponíveis em alguns casos, permitindo que o sistema operacional seja executado em hardware “nativo”, tudo nos limites de um computador e de um sistema operacional modernos.

O advento dos sistemas operacionais de código-fonte aberto também torna fácil fazer a transição de aluno para desenvolvedor de sistema operacional. Com algum conhecimento, algum esforço e uma conexão com a Internet, um aluno pode até mesmo criar uma nova distribuição de sistema operacional! Há apenas alguns anos, era difícil ou impossível obter acesso ao código-fonte. Agora esse acesso só é limitado quanto ao tempo e ao espaço em disco que o aluno possui.

*Fonte: (Fundamentos de Sistemas Operacionais – SILBERSCHATZ, Abraham S.; GALVIN, Peter Baer; GAGNE, Greg – 8ª Edição, 2012).*

### 3.3.7 SISTEMAS DE ARQUIVOS

[TANENBAUM, 2012], chega à conclusão que todos os aplicativos do computador precisam armazenar e recuperar informações. Enquanto um processo está em execução, ele pode armazenar um volume de informações limitado dentro de seu próprio espaço de endereçamento. Entretanto, a capacidade de armazenamento está restrita ao tamanho do espaço de endereçamento virtual. Para algumas aplicações, esse tamanho é adequado, mas para outras, como reservas de passagens aéreas, sistemas bancários ou registros corporativos, ele é pequeno demais.

Um segundo problema na manutenção de informações dentro do espaço de endereçamento de um processo é que, quando o processo termina, as informações são perdidas. Para muitas aplicações, (por exemplo, para bancos de dados), as informações devem ser mantidas por semanas, meses ou até

para sempre. É inaceitável perdê-las quando o processo que as está usando termina. Além disso, elas não devem desaparecer quando uma falha no computador elimina o processo.

Um terceiro problema é que, frequentemente, são necessários vários processos para acessarem as informações (partes delas) ao mesmo tempo. Por exemplo, se temos um catálogo telefônico on-line armazenado dentro do espaço de endereçamento de um único processo, apenas esse processo pode acessá-lo. A maneira de resolver esse problema é tornar as informações em si independentes de qualquer processo.

Assim, temos três requisitos fundamentais para o armazenamento de informações a longo prazo:

Deve ser possível armazenar um volume muito grande de informações.

As informações devem sobreviver ao término do processo que as estão utilizando.

Vários processos devem ser capazes de acessar as informações concomitantemente.

A solução usual para todos esses problemas é armazenar as informações em discos e outras mídias externas, em unidades chamadas arquivos. Então, os processos podem ler e escrever informações novas, se for necessário. As informações armazenadas em arquivos devem ser persistentes; isto é, não devem ser afetadas pela criação e pelo término do processo. Um arquivo só deve desaparecer quando seu criador o remover.

Os arquivos são gerenciados pelo sistema operacional. O modo como eles são estruturados, nomeados, acessados, usados, protegidos e implementados são tópicos importantes no projeto do sistema operacional. Como um todo, a parte do sistema operacional que trata com arquivos é conhecida como sistema de arquivos.

Do ponto de vista dos usuários, o aspecto mais importante de um sistema de arquivos é como ele aparece para eles; isto é, o que constitui um arquivo, como os arquivos recebem seus nomes e como são protegidos, quais operações são permitidas, etc. Os detalhes de serem utilizadas cadeadas ou mapas de bits para monitorar o espaço de armazenamento livre e de que existem muitos setores em um bloco lógico têm menos interesse, embora sejam de grande importância para os projetistas do sistema de arquivos.

- Atribuição de nomes de arquivo
- Estrutura de arquivo
- Tipos de arquivo
- Acesso a um arquivo
- Atributos de arquivo
- Operações sobre arquivos
- Diretórios
- Diretórios simples
- Sistemas de diretório hierárquicos
- Nomes de caminho
- Operações sobre diretórios

*Fonte: (TANENBAUM e WOODHULL – Sistemas operacionais – Projeto e Implementação – terceira edição - 2012).*

### 3.3.8 GERENCIAMENTO DE MEMÓRIA

Para [TANENBAUM, 2012], a memória é um recurso importante que deve ser cuidadosamente gerenciado. Embora, hoje em dia, um computador doméstico médio tenha duas mil vezes mais memória do que o IBM 7094 (o maior computador do mundo no início dos anos 60), os programas e os dados que eles devem manipular também cresceram tremendamente.

Teoricamente, o que todo programador gostaria é de uma memória infinitamente grande, infinitamente rápida e que também fosse não-volátil; isto é, que não perdesse seu conteúdo na falta de energia elétrica. E já que estamos nessa, por que não pedir também que fosse barata? Infelizmente, a tecnologia não consegue tornar esses sonhos uma realidade. Consequentemente, a maioria dos computadores tem uma hierarquia de memória, com uma pequena quantidade de memória cache, volátil, muito rápida e cara; centenas de megabytes de memória principal volátil (RAM) de velocidade e preços médios; e dezenas ou centenas de gigabytes de armazenamento em disco, não-volátil, lento e barato. A tarefa do sistema

operacional é coordenar a utilização desses diferentes tipos de memória.

A parte do sistema operacional que gerencia a hierarquia de memória é chamada de gerenciador de memória. Sua tarefa é monitorar as partes da memória que estão em uso e as que não estão, alocar memória para os processos quando eles precisarem dela e liberá-la quando terminam, e gerenciar a transferência (swapping) entre a memória principal e o disco, quando a memória principal for pequena demais para conter todos os processos. Na maioria dos sistemas o gerenciador de memória fica no núcleo.

Historicamente, a memória principal sempre foi vista como um recurso escasso e caro. Uma das maiores preocupações dos projetistas foi desenvolver sistemas operacionais que não ocupassem muito espaço de memória e, ao mesmo tempo, otimizassem a utilização dos recursos computacionais. Mesmo atualmente, com a redução de custo e consequente aumento da capacidade da memória principal, seu gerenciamento é um dos fatores mais importantes no projeto de sistemas operacionais.

Em geral, programas são armazenados em memórias secundárias, como disco ou fita, por ser um meio não volátil, abundantemente e de baixo custo. Como o processador somente executa instruções localizadas na memória principal, o sistema operacional deve sempre transferir programas da memória secundária para a memória principal antes de serem executados. Como o tempo de acesso à memória secundária é muito superior ao tempo de acesso à memória principal, o sistema operacional deve buscar reduzir o número de operações de E/S à memória secundária, caso contrário, podem ser ocasionados sérios problemas no desempenho do sistema.

A gerência de memória deve tentar manter na memória principal o maior número de processos residentes, permitindo maximizar o compartilhamento do processador e demais recursos computacionais. Mesmo na ausência de espaço livre, o sistema deve permitir que novos processos sejam aceitos e executados. Isso é possível através da transferência temporária de processos residentes na memória principal para a memória secundária, liberando espaço para novos processos. Esse mecanismo é conhecido como swapping.

*Fonte: (TANENBAUM e WOODHULL – Sistemas operacionais – Projeto e Implementação – terceira edição - 2012).*

Outra preocupação na gerência de memória é permitir a execução de programas que sejam maiores que a memória física disponível, implementada através de técnicas como o overlay e a memória virtual.

Em um ambiente de multiprogramação, o sistema operacional deve proteger as áreas de memória ocupadas por cada processo, além da área onde reside o próprio sistema. Caso um programa tente realizar algum acesso

indevido à memória, o sistema de alguma forma deve impedi-lo. Apesar de a gerência de memória garantir a proteção de áreas da memória, mecanismos de compartilhamento devem ser oferecidos para que diferentes processos possam trocar dados de forma protegida. [TAMENBAUM, 2012].

### 3.3.8.1 MEMÓRIA SÓLIDA PORTÁTIL NÃO-VOLÁTIL

Memória que pode reter dados quando a energia é desligada é uma parte essencial de um sistema eletrônico moderno. [SAKUI, Koji e SUH, Kang-Deog, 2008, p. 233].

Conforme publicado por, [MORIMOTO, 2007], na revista Guia do Hardware, onde a informação permanece mantida em publicação no Livro Hardware II – O Guia Definitivo – MORIMOTO, Carlos Eduardo, 2013. 1086p.

Diferentemente da memória RAM e também das SRAM, a memória Flash permite armazenar dados por longos períodos, sem precisar de alimentação elétrica. Graças a isso, a memória Flash se tornou rapidamente a tecnologia dominante em cartões de memória, pendrives, HDs de estado sólido (SSDs), memórias de armazenamento em câmeras, celulares e palmtops e assim por diante.

Se a memória Flash não existisse, todas essas áreas estariam muito atrasadas em relação ao que temos hoje. Os celulares e os palmtops provavelmente ainda utilizariam memória SRAM para armazenar os dados e seriam por isso mais caros e perderiam os dados quando a bateria fosse removida. Os pendrives simplesmente não existiriam e os cartões de memória estariam estagnados nos cartões compact-flash, utilizando microdrives ou pequenas quantidades de memória SRAM alimentada por uma pequena bateria. Formatos mais compactos, como os cartões SD e miniSD simplesmente não existiriam.

*Fontes: (Carlos E. Morimoto. Memória flash Revista Guia do Hardware – Ano 1 – Nº 4 – Abril/2007 - Visitado em 4 de Julho de 2014). Disponível também em: Livro Hardware II – O Guia Definitivo – Morimoto, Carlos Eduardo – Porto Alegre: Sul Editores, 2013. 1086p. / ISBN 978-85-99593-16-5 / www.gdhpress.com.br – Disponível em 11/2014 – Biblioteca Central da UTFPR.*



#### **4. ANÁLISE DOS DADOS**

Sabemos que a abrangência deste trabalho de pesquisa é um tanto quanto limitada, a mesma aborda uma quantidade pequena de indivíduos de diversas áreas, porém, podemos ter uma ideia de como seria caso tenhamos um produto em mãos, já configurado e pronto para uso.

O trabalho acadêmico envolve a fusão de hardware e software, ou seja, um sistema operacional atrelado a uma memória sólida portátil não-volátil, definido em um só produto, o qual, somado a pesquisa realizada em campo, a qual nos reportou uma aceitação de forma unânime por parte dos entrevistados, chegamos a ideia que, podemos utilizá-lo como um produto, o qual poderá ocorrer uma suposta distribuição em massa, sendo direcionado para qualquer tipo de público interessado. Uma memória sólida portátil não-volátil que, além das suas funcionalidades habituais, como: armazenagem dos dados, gravações, entre outras, acaba por atribuir também um forte valor agregado ao dispor de um recurso a mais, ou seja, a inicialização de um sistema operacional qualquer escolhido pela preferência do usuário.

Por fim, analisando os resultados obtidos, mencionamos a possibilidade de base para uma patente na área.

## 5. CONSIDERAÇÕES FINAIS

Temos muito a estudar e desenvolver para obter uma melhor comunicação, convergir sistemas operacionais aos mais diversos hardwares, ligá-los em rede, via software ou ainda entre máquinas, buscar padrões os quais possam interagir com todos os tipos de usuários em meio a tantos padrões, tarefa à qual nos proporciona grandes desafios em nosso cotidiano.

Muitas vezes deixamos de utilizar soluções, pois partimos da ideia que é difícil, complexa, insegura e ineficiente, quando na verdade é totalmente ao contrário, com isso obtemos uma quebra de paradigma no sentido da resistência às mudanças.

O trabalho adiciona impacto no avanço tecnológico em confronto direto com a gestão da tecnologia da informação e comunicação, oferecendo melhores práticas ao manuseio dos hardwares, softwares e sistemas operacionais, mais precisamente atribuído a falta de planejamento como um todo, envolvendo nos processos: tecnologias da informação e comunicação, pessoas e sistema de hardware e software.

O conhecimento obtido com este trabalho é de grande utilidade para diversos públicos, os quais podem melhorar a gestão da tecnologia da informação e comunicação no seu dia-a-dia. Apesar de nenhum dos entrevistados dispor no momento da entrevista de um sistema operacional pronto para ser utilizado em sua memória sólida portátil não-volátil, concordou de forma unânime com a ideia, que, ao ter um sistema operacional em sua memória sólida portátil não-volátil pronto para ser utilizado, o mesmo pode fazer toda a diferença quando lhe for necessário ou oportuno a utilização deste recurso ou acesso em seu dia-a-dia.

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

MORIMOTO, Carlos Eduardo – *Dicionário Técnico de Informática 3ed* - Disponível em <http://www.guiadohardware.net> – Acesso em 21/02/2015.

MORIMOTO, Carlos Eduardo, 2013. *Livro Hardware II – O Guia Definitivo* – Porto Alegre: Sul Editores, 2013. 1086p. / ISBN 978-85-99593-16-5 / [www.gdhpress.com.br](http://www.gdhpress.com.br) – Acesso em 21/02/2015 – Disponível em 11/2014 na Biblioteca Central da UTFPR.

TANENBAUM, Andrew S. 2009. *Livro Sistemas Operacionais Modernos* / Tradução Ronaldo A. L. Gonçalves, Luís A. Consularo, Luciana do Amaral Teixeira; revisão técnica Raphael Y. de Camargo. – 3. Ed. – São Paulo: Pearson Prentice Hall, 2009. ISBN 978-85-7605-237-1 – Disponível em 11/2014 – Biblioteca Central da UTFPR.

TANENBAUM, Andrew S.; WOODHULL, Albert S. 2008. *O Livro do MINIX – Sistemas Operacionais – Projeto e Implementação* - Tradução João Tortello - Terceira Edição – Porto Alegre: Bookman, 2008. 922p.; 25cm. ISBN 978-85-7780-057-5 – Disponível em 11/2014 – Biblioteca Central da UTFPR.

SILBERSCHATZ, Abraham; GALVIN, Peter Baer; GAGNE, Greg. *Livro Fundamentos de Sistemas Operacionais* / Tradução: Aldir José Coelho da Silva; Revisão técnica Elisabete do Rego Lins. – 8. Ed. – Rio de Janeiro: LTC, 2012. ISBN 978-85-216-1747-1 – Disponível em 11/2014 – Biblioteca Central da UTFPR.

MACHADO, Francis Berenger; MAIA, Luiz Paulo. *Livro Arquitetura de Sistemas Operacionais - 5. Ed.* – Rio de Janeiro: LTC, 2013. ISBN 978-85-216-2210-9 – Disponível em 11/2014 – Biblioteca Central da UTFPR.

MAZIERO, Carlos Alberto. *Livro em forma de e-book - Sistemas Operacionais: Conceitos e Mecanismos* – DAINF – UTFPR – Prof. Carlos Alberto Maziero, 2013 – Disponível em <http://dainf.ct.utfpr.edu.br/~maziero/lib/exe/fetch.php/so:so-livro.pdf> – Acesso em 21/02/2015.

MAZIERO, Carlos Alberto. *Uma breve história do UNIX, LINUX e Software Livre* - DAINF – UTFPR - Prof. Carlos Alberto Maziero, 2013 – Disponível em [http://dainf.ct.utfpr.edu.br/~maziero/doku.php/unix:historico\\_do\\_unix\\_e\\_linux](http://dainf.ct.utfpr.edu.br/~maziero/doku.php/unix:historico_do_unix_e_linux) – Acesso em 21/02/2015.

LINHARES, Robson Ribeiro. Dissertação de Mestrado – UTFPR - *Modelamento de Hardware Visando À Estimativa do Tempo de Execução de Programas* – Disponível em - [http://files.dirppg.ct.utfpr.edu.br/cpgei/Ano\\_2001/dissertacoes/CPGEI\\_Dissertacao\\_228\\_2001.pdf](http://files.dirppg.ct.utfpr.edu.br/cpgei/Ano_2001/dissertacoes/CPGEI_Dissertacao_228_2001.pdf) - Acessado em 21/02/2015.

EPOS – *Um Sistema Operacional Portável para Sistemas Profundamente Embarcados* – (Hugom, Arliones, Lucas, Cancian, Danillo, Guto) – Laboratório de Integração de Software e Hardware – UFSC - Anais do XXVI Congresso da SBC – WSO – III Workshop de Sistemas Operacionais – 14 a 20 de julho de 2006 – Campo Grande, MS.

*Portabilidade de Sistemas operacionais no domínio de sistemas embarcados* – Laboratório de Integração de Software e Hardware – UFSC – (Hugom, Arliones, Lucas, Guto) 2006. – Financiado pela FINEP – Financiadora de Estudos e Projetos. Disponível em [http://www.lisha.ufsc.br/pub/Marcondes\\_CLEI\\_2006.pdf](http://www.lisha.ufsc.br/pub/Marcondes_CLEI_2006.pdf) - Acessado em 21/02/2015.

USP – Escola de Engenharia de São Carlos - Projeto de Iniciação Científica – *Automação usando Sistemas Operacionais Linux Embarcados em Microcontroladores ARM* – 2011. Disponível em <https://uspdigital.usp.br/siicusp/cdOnlineTrabalhoVisualizarResumo?numeroInscricaoTrabalho=3864&numeroEdicao=20> – Acesso em 21/02/2015.

Revista de Informática Teórica e Aplicada – RITA – Volume VIII – Número 3 – Dezembro 2001. Disponível em <http://www.lume.ufrgs.br/bitstream/handle/10183/19242/000102159.pdf> - Acessado em 21/02/2015.

Revista Guia do Hardware – ANO 1 - Nº 4 – Abril – 2007 – Especial Memórias. Disponível em [http://e.cdn-hardware.com.br/static/media/RevistaGDH\\_04.pdf](http://e.cdn-hardware.com.br/static/media/RevistaGDH_04.pdf) - Acessado em 21/02/2015.

INTEL, Flash Memory, 1994 – Vol.1 – ISBN – 1-55512-200-0 - disponível em (Biblioteca Copel) – Acessado em 11/2014.

INTEL, Flash Memory, 1994 – Vol.2 – ISBN – 1-55512-214-0 - disponível em (Biblioteca Copel) – Acessado em 11/2014.

Artigo Legado da Apple sobre os recursos USB em Macs com gráficos AGP – Última modificação em 20/02/2012 - Disponível em <http://support.apple.com/kb/TA25908> - Acessado em 22/02/2015.

Artigo IBM – MICHAEL, Singer. 2005 - IBM brains capture a PC's soul – Disponível em [http://news.cnet.com/IBM-brains-capture-a-PCs-soul/2100-1041\\_3-5830870.html](http://news.cnet.com/IBM-brains-capture-a-PCs-soul/2100-1041_3-5830870.html) - Acessado em 21/02/2015.

*Nonvolatile Memory Technologies with Emphasis on Flash*. Edited by J. E. Brewer and M. Gill - Copyright 2008. The Institute of Electrical and Electronics Engineers, Inc. Online ISBN: 9780470181355 – Disponível no acervo Online IEEE – Banco de dados – UTFPR – Acesso em 11/2014.

SAKUI, Koji e SUH, Kang-Deog, 2008 p. 223 – *Nonvolatile Memory Technologies with Emphasis on Flash*. Edited by J. E. Brewer and M. Gill - Copyright 2008. The Institute of Electrical and Electronics Engineers, Inc. Online ISBN: 9780470181355 – Disponível no acervo Online IEEE – Banco de dados – UTFPR – Acesso em 11/2014.