Marisangila Alves, MSc
marisangila.alves@edu.sc.senai.br
marisangila.com.br

2026/1

Fundamentos de Desenvolvimento de Sistemas

Noções Básica de Hardware

Sumário

Software

As vezes o chamamos de programa, aplicativo, aplicação, sistema...

Software

Software é o conjunto de programas, instruções e dados que controlam o funcionamento do hardware e permitem que o computador se torne funcional para executar tarefas.

- O **hardware** representa os componentes físicos do computador.
- O **software** corresponde às instruções que dizem ao hardware o que fazer, ou seja, a lógica.
- Sem software, o hardware não executa nenhuma tarefa útil.

Classificação

- **Software de Base** – controla o funcionamento do computador e gerencia os recursos de hardware.
- **Software Aplicativo** – programas utilizados pelos usuários para realizar tarefas específicas.

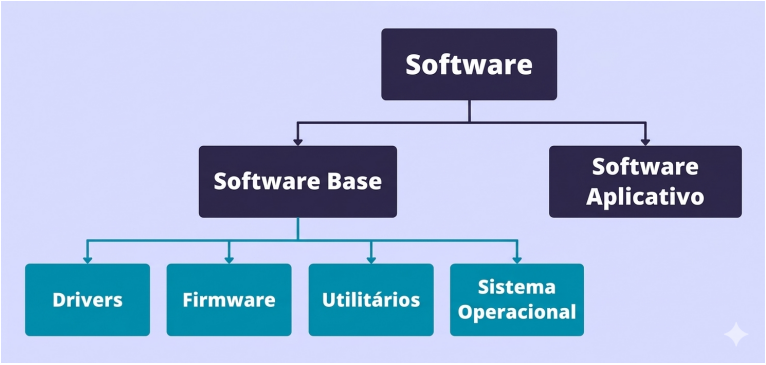

Figura 1: Classificação de software. Imagem gerada por IA.

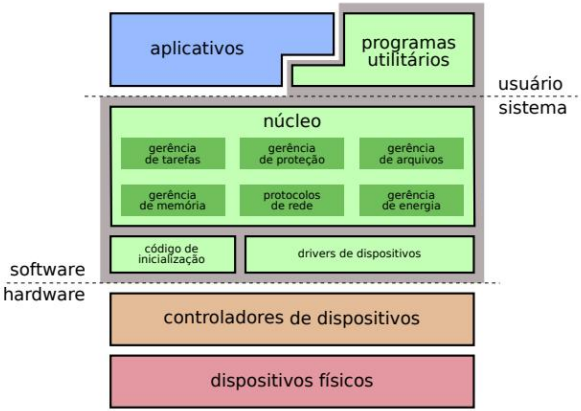

Figura 2: Camadas de software (Maziero, 2019).

Software de Base

Software de Base

Conjunto de programas responsáveis por gerenciar o hardware do computador e fornecer serviços para outros software aplicativos.

Exemplos:

- Windows;
- Drivers de placa de vídeo;
- Menu de Configurações;
- BIOS;
- Gerenciador de arquivos;
- Android.

Sistemas Operacionais

Sistema Operacional

O sistema operacional é o software responsável por gerenciar os recursos do computador e fornecer uma interface entre o usuário, os programas e o hardware.

Principais funções do sistema operacional:

- Gerenciamento de processos;
- Gerenciamento de memória;
- Gerenciamento de dispositivos;
- Gerenciamento de arquivos;
- Interface com o usuário.

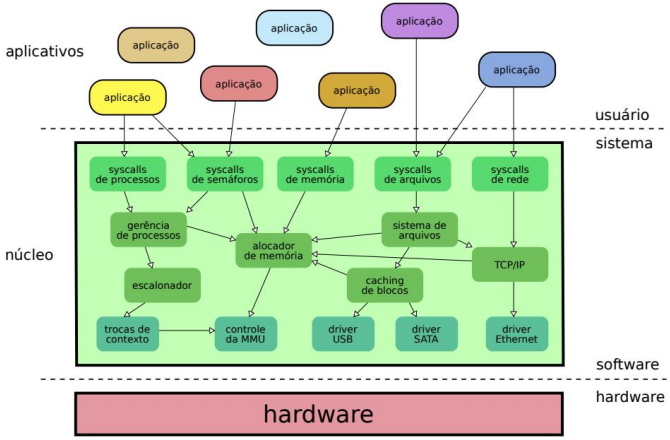

Figura 3: Arquitetura simplificada de um sistema operacional (Maziero, 2019).

Drivers

Driver

Driver é um software que permite que o sistema operacional se comunique e controle dispositivos de hardware específicos.

Exemplos:

- Driver de placa de vídeo;
- Driver de impressora;
- Driver de rede;
- Driver de áudio.

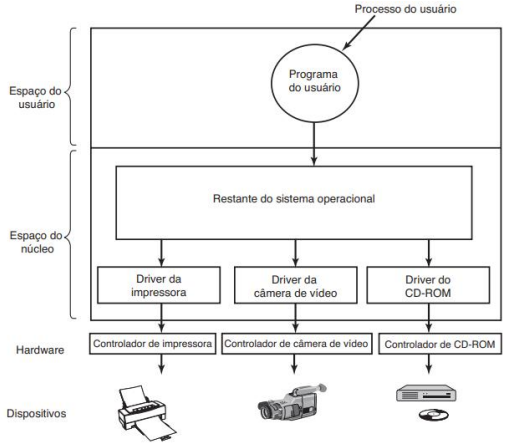

Figura 4: Comunicação entre hardware, drivers e sistema operacional (TANENBAUM, 2010).

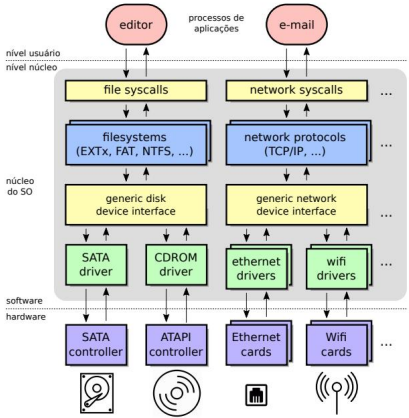

Figura 5: Drivers no sistema operacional (Maziero, 2019).

Firmware

Firmware

Firmware é um tipo de software gravado permanentemente em memória não volátil (que pode ser somente de leoitura (ROM)) de um dispositivo eletrônico, responsável por tornar o hardware funcional.

Exemplos:

- BIOS / UEFI;
- firmware de roteadores;
- firmware de microcontroladores de dispositivos embarcados em geral;
- Que podem ser: relógios digitais, contorle de TV, ar condicionado, microondas, geladeira, câmera.

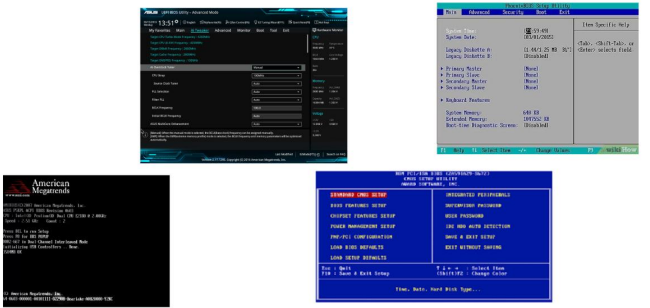

Figura 6: Exemplo de firmware (BIOS/UEFI).

Software Aplicativo

Aplicativos

Software aplicativo é composto por programas que permitem ao usuário realizar tarefas específicas utilizando o computador.

Exemplos:

- Navegadores web;
- Editores de texto;
- Planilhas eletrônicas;
- Sistemas industriais e de automação;
- Sistemas corporativos;
- Editores de aúdio, vídeo;
- Player de música e vídeo;
- Visualizar de PDF;
- Jogos.

Camadas de um sistema computacional I

- **Hardware**: componentes físicos do computador.
- **Firmware**: inicializa e controla dispositivos.
- **Sistema Operacional**: gerencia os recursos do sistema.
- **Aplicativos**: programas utilizados pelos usuários.

Figura 7: Camada de Software sob Hardware. Imagem gerada por IA.

Licenças de Software

Licença de Software

Uma licença de software é um acordo legal que define como um programa pode ser utilizado, copiado, modificado ou distribuído pelos usuários.

- **Software Proprietário**: o código-fonte não é público e seu uso é restrito pelo fabricante.
  - Exemplos: Windows, Microsoft Office.
- **Software Livre**: permite executar, estudar, modificar e redistribuir o programa.
  - Exemplos: GNU/Linux, LibreOffice.
- **Software Open Source**: código-fonte disponível para estudo e modificação, geralmente com colaboração da comunidade.
  - Exemplos: Python, Firefox.

Licença	Características
GPL (GNU General Public License)	Permite uso, modificação e distribuição, exige que derivados sejam livres.
MIT License	Muito permissiva; permite reutilização do código com poucas restrições.
Apache License	Permissiva e inclui proteção contra patentes.
BSD License	Similar à MIT, com poucas restrições de uso.
Creative Commons	Usada para obras como textos, imagens e vídeos.

Tabela 1: Exemplos de licenças populares de software e conteúdo digital

Projeto GNU

O **Projeto GNU** foi iniciado em 1983 por **Richard Stallman** com o objetivo de desenvolver softwares completamente livre, no qual os usuários tivessem liberdade para executar, estudar, modificar e compartilhar o software.

O sistema operacional amplamente conhecido como **Linux** é, na realidade, uma combinação do **kernel Linux** com as ferramentas do **Projeto GNU**, formando o sistema **GNU/Linux**.

Richard Stallman

GNU

Logo do Projeto GNU

Compilador

Compilação

Compilar é o processo de traduzir um programa escrito em uma linguagem de programação (código fonte) para linguagem de máquina antes da execução. Esse processo é realizado por um **compilador**, que gera um arquivo executável que pode ser executado diretamente pelo computador.

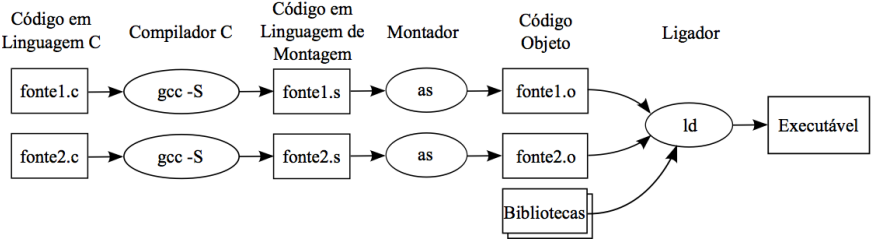

Figura 8: Processo de execução de um programa.

- **Montador**: programa que traduz a **linguagem de montagem** para **código de máquina**.
- **Linguagem de montagem**: representação simbólica da linguagem de máquina de um processador específico, acrescida de instruções adicionais que facilitam a escrita do programa e fornecem diretivas para o montador.
- **Compilador**: programa que converte um código escrito em uma **linguagem de programação** (linguagem fonte) para **linguagem de máquina** (código objeto). Pode oferecer recursos como variáveis automáticas, expressões aritméticas, estruturas de controle (`FOR`, `WHILE`), escopo de variáveis e funções de alto nível.

- **Código executável**: código de máquina gerado por um compilador ou montador, pronto para ser executado pelo computador.
- **Conjunto de instruções**: conjunto de todas as instruções de linguagem de máquina que um determinado processador é capaz de executar.
- **Ligador (Linker)**: programa utilitário que combina um ou mais arquivos de **código objeto** gerados separadamente em um único arquivo executável.

- **Carregador (Loader)**: rotina que carrega um programa executável na memória para que ele possa ser executado.
- **Linguagem de máquina (código de máquina)**: representação binária de um programa que é diretamente interpretada e executada pelo processador.
- **Código objeto**: versão em linguagem de máquina do código fonte gerada por um compilador ou montador, que posteriormente é transformada em código executável pelo ligador.

Interpretação

Interpretar é executar um programa traduzindo suas instruções passo a passo durante a execução. Nesse caso, um **interpretador** lê o código fonte e executa cada instrução diretamente, sem gerar um arquivo executável previamente.

Compilador traduz todo o programa antes da execução; interpretador traduz e executa o programa passo a passo durante a execução.

Característica	Compilador	Interpretador
Processamento	Traduz o programa inteiro antes de executar	Traduz e executa ao mesmo tempo
Arquivo executável	Gera executável	Não gera executável
Ferramentas auxiliares	Usa linker e loader	Não usa linker
Desempenho	Execução mais rápida	Execução mais flexível

Tabela 2: Comparação entre compiladores e interpretadores

(Stallings, 2017)

Referências I

DE SOUZA, M.A.F. *et al.* **Algoritmos e Lógica de Programação**. São Paulo: Thomson Learning, 2004.

FREE SOFTWARE FOUNDATION. **O Projeto GNU**. Acesso em: 16 mar. 2026. Projeto GNU. 2025. Disponível em: `https://www.gnu.org/gnu/thegnuproject.pt-br.html`.

MAZIERO, Carlos Alberto. **Sistemas Operacionais: Conceitos e Mecanismos [recurso eletrônico]**. Curitiba: DINF - UFPR, 2019. ISBN 978-85-7335-340-2.

STALLINGS, William. **Arquitetura e Organização de Computadores**. Última edição. [*S. l.*]: Pearson, 2017.

TANENBAUM, Andrew S. **Sistemas Operacionais Modernos**. 3. ed. São Paulo: Pearson, 2010.

Estes slides estão protegidos por uma licença Creative Commons

Este modelo foi adaptado de Maxime Chupin.