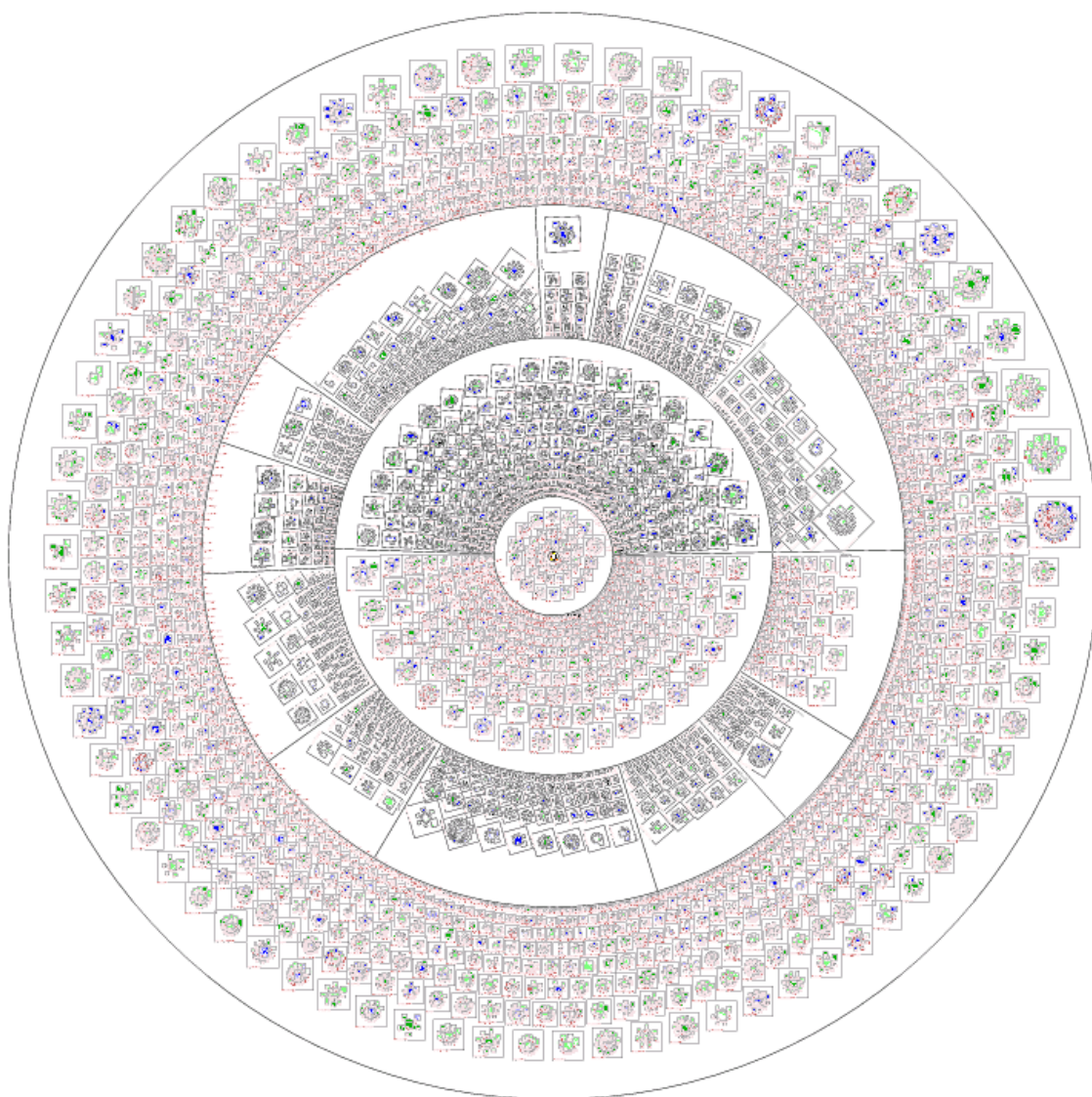


Kernel Linux



Representação artística do núcleo Linux

Trabalho Realizador Por:

Tiago Conceição Nº 11903



Índice

Introdução.....	2
O que é o Kernel.....	3
Como surgiu.....	4
Para que serve.....	5
Versões.....	6
Versões iniciais.....	6
Versões estáveis.....	6
Processo de Inicialização do Kernel.....	8
Diferenças Kernel Windows x Linux.....	10
Conclusão.....	11



Introdução

Neste trabalho sobre o Kernel do Linux será exposto os seguintes tópicos:

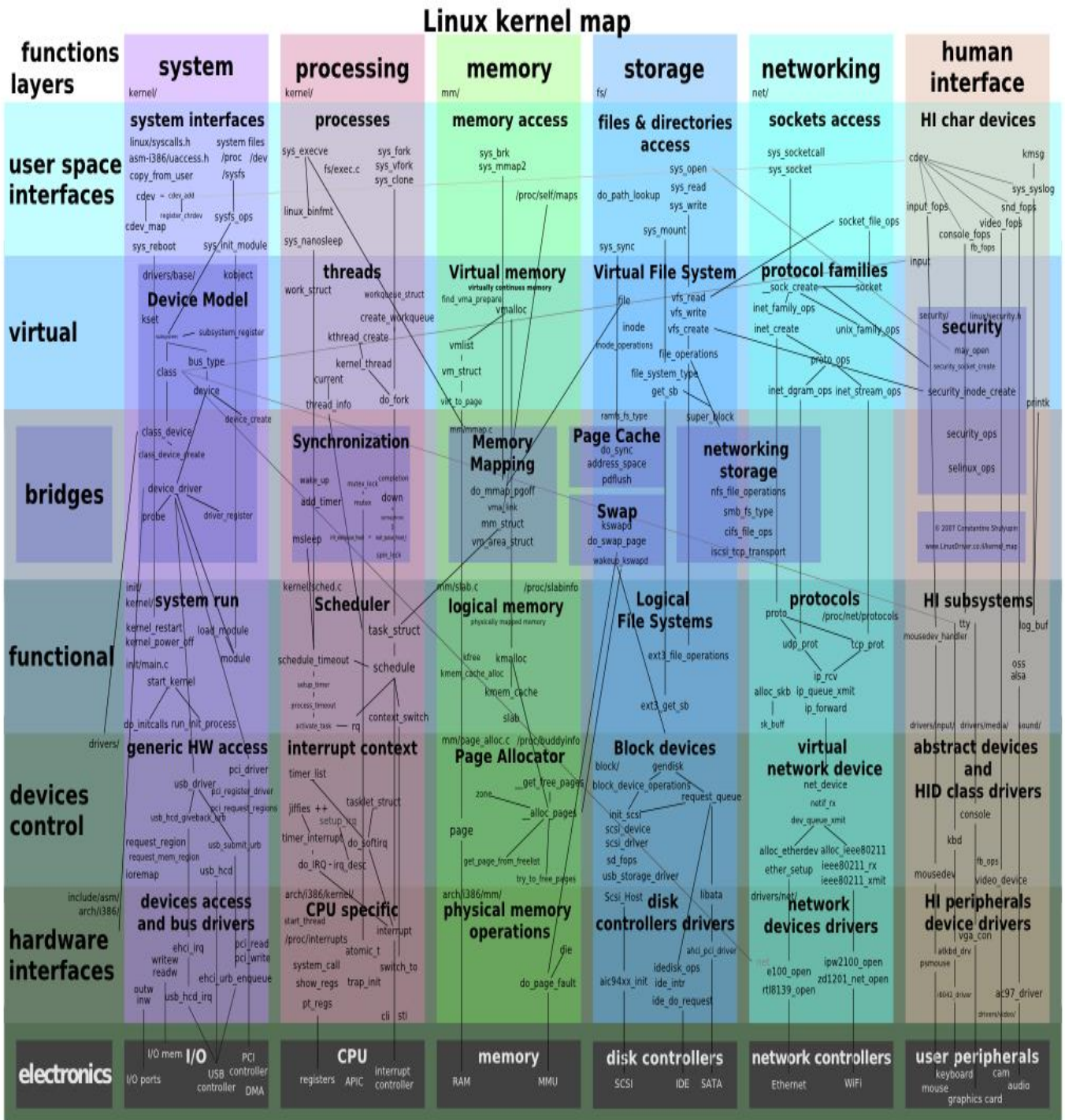
1. O que é o Kernel
2. Como surgiu
3. Para que serve
4. Versões
5. Processo de Inicialização do Kernel
6. Diferenças Kernel Windows x Linux

Esta pesquisa tem por meio “contar” um pouco da história do Linux, comparar com o actual Microsoft Windows, e como este se tornou importante nos dias de hoje.



O que é o Kernel

O **Kernel** é o núcleo do sistema operativo. Cabe ao Kernel fazer a ligação entre o hardware e os programas executados pelo computador. A junção do Kernel mais o software é que tornam o computador usável (drivers, protocolos de comunicação, entre outros), de acordo com a sua aplicação, formam o sistema operativo em si.





Como surgiu

Linus Torvalds, com cerca de 22 anos, decidiu criar a sua própria implementação de um terminal no seu computador 80386, principalmente para aceder ao servidor UNIX da instituição de ensino. Isso porque ele já tinha testado o MINIX para esta finalidade, mas não estava satisfeito com os seus recursos.

A intenção de Torvalds era fazer com que o projecto rodasse especificamente na sua máquina 80386, sendo o desenvolvimento feito a partir do MINIX. O trabalho avançou de tal forma que chegou um ponto onde Torvalds já tinha um Kernel funcional.

Em 1991, Linus Torvalds decidiu divulgar abertamente o seu projecto. Para isso, publicou mensagens na Usenet (uma espécie de antecessora da internet, baseada em troca de mensagens) pedindo sugestões e colaborações para a sua iniciativa.

O início da trajectória do Linux não foi isento de problemas. Uns dos obstáculos que Torvalds teve que enfrentar foram as críticas do professor Andrew S. Tanenbaum, em que afirmou que o "Linux é obsoleto", especialmente por este ter um "design monolítico".

Tanenbaum não estava contente com o facto de o Linux ter sido preparado especificamente para funcionar com o processador 80386 que, além de caro, teria sua arquitectura substituída futuramente, o que, na verdade, não aconteceu.

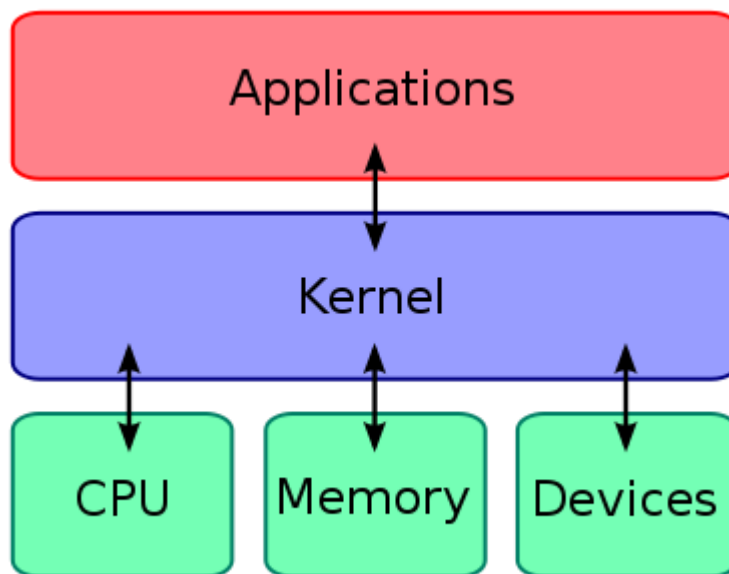
Linus respondeu às críticas e continuou o seu trabalho, cada vez mais contando com o apoio de mais pessoas. Com o passar do tempo, o Linux acabou por ser portado para várias plataformas, o que certamente contribuiu para o seu sucesso.





Para que serve

Cabe ao Kernel fazer a ligação entre o hardware e os programas executados pelo computador. A junção do Kernel mais o software é que tornam o computador usável (drivers, protocolos de comunicação, entre outros), de acordo com a sua aplicação, formam o sistema operativo em si.





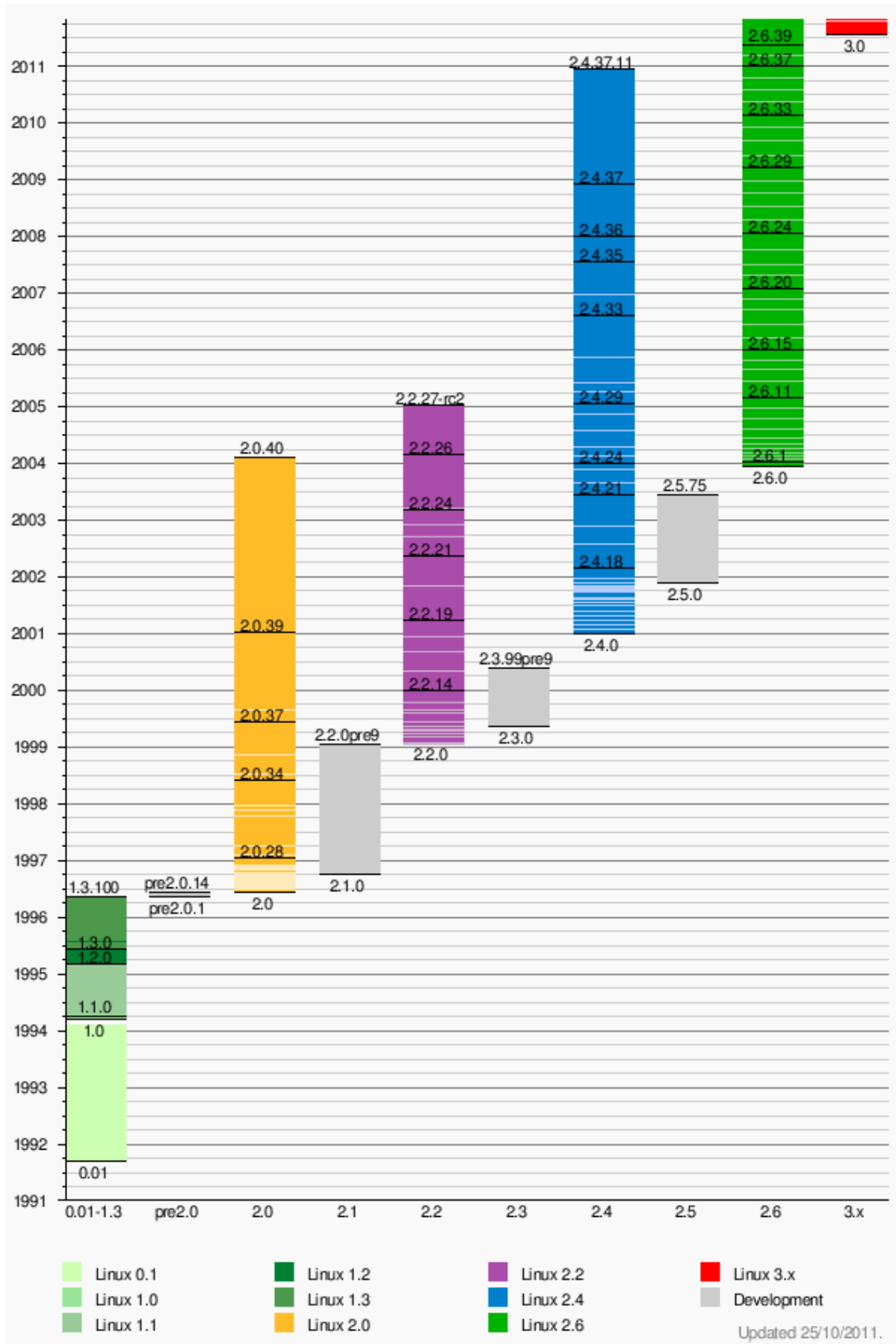
Versões

Versões iniciais

- **0.01** - Setembro de 1991 - primeira versão (10.239 linhas de código).
- **0.02** - Outubro de 1991 - primeira versão disponibilizada publicamente.
- **0.12** - Janeiro de 1992 - primeira versão sob GPL (19.258 linhas de código).

Versões estáveis

- **1.0** - Março de 1994 - Suportava apenas máquinas monoprocessadoras i386 (176.250 linhas de código).
- **1.2** - Março de 1995 - Adicionado suporte para Alpha, SPARC e MIPS (310.950 linhas de código).
- **2.0** - Junho de 1996 - Adicionado suporte para mais processadores e foi incluído suporte para SMP (777.956 linhas de código).
- **2.2** - Janeiro de 1999 (1.800.847 linhas de código).
- **2.4** - Janeiro de 2001 (3.377.902 linhas de código).
- **2.6** - Dezembro de 2003 (5.929.913 linhas de código).



Linha do tempo



Processo de Inicialização do Kernel

Inicializar o Sistema

- O processo de inicialização de sistemas é independente do sistema operativo e depende muito do tipo de hardware em uso.
- No caso de PCs baseados na arquitetura x86 o processo de inicialização (boot) inicia-se na BIOS, no endereço 0xFFFF0. O primeiro passo da BIOS é executar o Power On Self Test (POST) que tem como principal função a verificação do hardware e a enumeração e inicialização dos dispositivos locais.
- Para inicializar o sistema operacional, a BIOS irá então procurar por dispositivos activos e válidos que estejam listados (por ordem de preferência) na CMOS (boot device). Os dispositivos de boot podem ser floppy disks, hard disks, flash drivers, dispositivos de rede, etc.
- De maneira geral, o Linux é inicializado a partir de um disco rígido, onde o MBR (Master Boot Record) contém o boot loader, e não o kernel. O MBR é apenas um setor de 512 bytes, localizado no cilindro 0, setor 1 do disco. Portanto no MBR apenas reside o suficiente para carregar o primeiro estágio do boot loader.

Boot Loader - Estágio 1

- O primeiro estágio do boot loader, ou primary boot loader, consiste do executável do primeiro estágio propriamente dito e uma pequena tabela de partições. A única função do primary boot loader é encontrar o secondary boot loader, ou o segundo estágio do boot loader, em uma partição ativa dentre as partições listadas e carregá-lo na memória.

Boot Loader - Estágio 2

- O segundo estágio do boot tem como função carregar o kernel do Linux na memória.
- Os 2 estágios do boot loader combinados formam o LILO (Linux Loader) ou o GRUB (GRand Unified Bootloader). O GRUB é um gerenciador de boot mais recente, que corrigiu algumas das falhas do LILO.
- Neste estágio é possível ver e escolher entre diversos tipos de kernel instalados (definidos em /boot/grub/menu.lst), e passar parâmetros de boot durante a inicialização do sistema.
- Quando o segundo estágio estiver carregado na memória o sistema de ficheiros é consultado, a imagem do kernel e a imagem initrd são carregados na memória, e quando esse processo é concluído, o boot loader invoca o kernel.

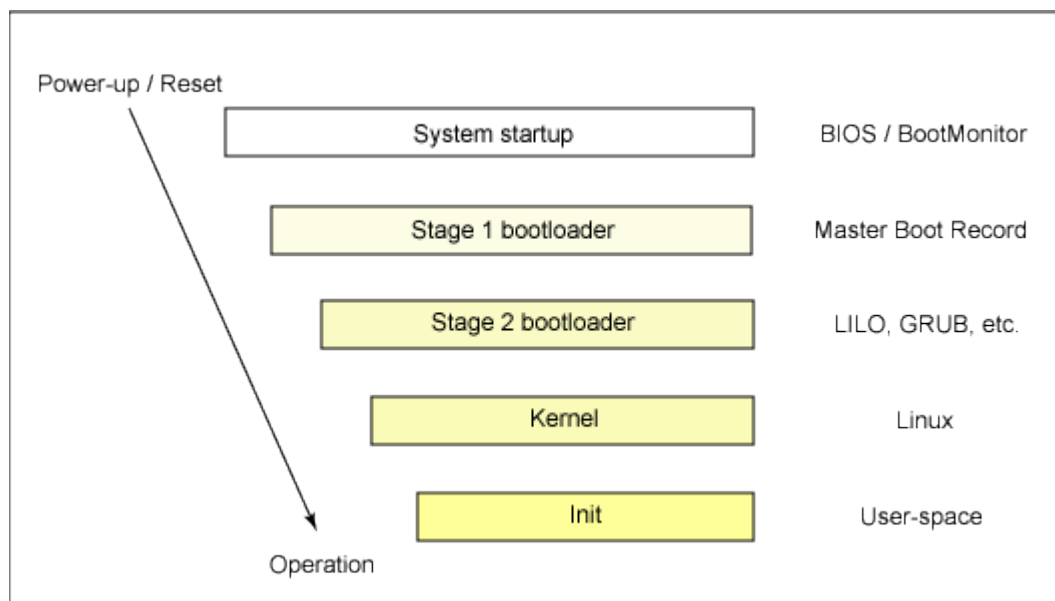


Kernel

- A imagem do kernel não é propriamente um executável, porém contém uma série de rotinas que permitem descomprimir o kernel e carregá-lo na memória.
- Esse mecanismo permite que tenhamos imagens de kernel pequenas, e device drivers compilados como módulos e carregados, quando necessário, através de um arquivo que pode ser montado como um sistema de ficheiros existente apenas na memória. Tal artifício permite que o kernel seja completamente carregado, sem ter que aceder a um disco rígido. Numa etapa futura da carga do kernel, esse sistema de ficheiros temporários será substituído pelo root file system padrão.
- Esse mecanismo garante uma grande flexibilidade a sistemas Linux uma vez que permite que o próprio root file system, resida num servidor remoto (NFS).

O Processo INIT

- Após executar o boot e iniciar, o kernel inicia um programa (init), que irá iniciar tudo o resto para que o sistema fique funcional.
- Num sistema Linux de desktop, a primeira aplicação iniciada geralmente é o **/sbin/init**. Mas não obrigatoriamente.





Diferenças Kernel Windows x Linux

Tópico	Linux	Microsoft Windows
Preço	A maior parte das distribuições são disponibilizadas gratuitamente, mas existem umas pagas com um preço menor à do Microsoft Windows.	De 50€ a 300€ cada licença.
Acessibilidade	Embora a maioria das distribuições tenha melhorado dramaticamente na facilidade de uso, o Windows ainda é muito mais fácil de usar para novos usuários de computador.	A Microsoft fez diversos avanços e mudanças que proporcionou um sistema operativo muito mais fácil de usar, e embora não seja o sistema operativo mais fácil, ainda é mais fácil do que Linux.
Fiabilidade	A maioria das distribuições do Linux são notoriamente mais fiáveis e muitas vezes podem funcionar por meses e anos sem precisar de ser reiniciado.	Embora a Microsoft Windows tenha feito grandes melhorias em termos de fiabilidade sobre as últimas versões do Windows, este ainda não consegue competir com a fiabilidade do Linux.
Aplicações	O Linux tem uma grande variedade de programas disponíveis, utilitários e jogos. No entanto, o Windows tem uma selecção muito maior de aplicações e jogos disponíveis.	Por causa da grande quantidade de utilizadores do Microsoft Windows, há uma selecção muito maior de programas disponíveis, utilitários e jogos, onde a maior parte não existe para Linux.
Custo das Aplicações	Muitos dos programas disponíveis, utilitários e jogos são de código livre ou aberto. Mesmo os programas complexos como o Gimp, OpenOffice, StarOffice, e o Wine estão disponíveis gratuitamente ou a baixo custo.	Embora o Windows tenha uma grande variedade de programas, utilitários e jogos de graça, a maioria dos programas mais modernos e complexos vai custar entre 20€ e 200€ por cópia.
Hardware	Há empresas e fabricantes de hardware que têm feito grandes avanços no suporte de hardware, e hoje o Linux irá suportar a maioria dos dispositivos. No entanto, há muitas empresas que ainda não oferecem drivers ou suporte para o seu hardware em Linux.	Por causa da quantidade de usuários do Microsoft Windows e com o apoio mais amplo de drivers, o Windows tem um suporte muito maior para dispositivos de hardware e uma boa maioria dos fabricantes irá apoiar os seus produtos no Microsoft Windows.
Segurança	Linux é, e sempre foi um sistema operativo muito seguro. Embora ainda seja possível ser atacado, quando comparado ao Windows, é muito mais seguro.	Embora a Microsoft tenha feito grandes melhorias ao longo dos anos com a segurança do sistema operativo, ele continua a ser dos mais vulneráveis a vírus e outros ataques.
Código Aberto	Muitas das distribuições do Linux e programas são de código aberto o que permite aos utilizadores personalizar ou modificar o código como desejarem.	O Microsoft Windows não é código aberto e a maior parte dos programas também não.
Suporte	Embora seja mais difícil de encontrar utilizadores familiarizados com todas as distribuições de Linux, há uma vasta quantidade de documentação disponível online e ajuda, livros, e apoio.	O Microsoft Windows inclui a sua própria secção de ajuda (F1), e tem uma grande quantidade de documentação disponível online, assim como livros sobre cada uma das versões do Windows.



Conclusão

Quando Linus Torvalds criou o Linux, não imaginava que este projecto fosse ser tão importante no futuro. A intenção dele era criar um sistema para o seu próprio computador, para aceder a um servidor UNIX, mas que acabou por se tornar no que é hoje.

Linux é um sistema livre UNIX que pode ser usado não só em computadores como outros equipamentos: routers, telemóveis, consolas, etc.

Actualmente existe o projecto **Embedded Linux** que visa a integração do Linux em dispositivos, que usamos dia a dia tais como MP4, electrodomésticos, GPS, routers e outro tipo de equipamentos.

A base do Linux (Kernel) sendo livre, é possível modificar e melhorar. Ao contrário do código fechado onde ninguém pode tocar. As vantagens são principalmente o uso grátis e como é livre há comunidades a ajudar na construção e optimização do núcleo.

Estando o Kernel dividido por módulos torna este mais rápido e mais optimizado, ao contrário do Kernel do Windows que além de ter avançado muito ainda não é tão rápido e bloqueia mais facilmente.

Além da evolução do Linux em termos gráficos e o uso de janelas ainda não é tão fácil de utilizar como o Windows, tal como ainda a falta de suporte em Hardware para Linux por parte de alguns fabricantes.

Sendo os pontos fortes do Linux o **preço em geral, fiabilidade, segurança e o código aberto** é indicado para quem quer aprender algo novo para além do Windows, ou para trabalhar, onde não se tem que preocupar com faltas de desempenho ou vírus. Por vezes é necessário formatar o Windows de X em X tempo, enquanto o Linux pode correr durante anos sem ter que o reinstalar. Também é muito usado em servidores mais do que o próprio Windows, além das licenças do Windows serem muito caras o sistema em si não é bom e por vezes uma cosia mal configurada pode “encalhar” o sistema todo não deixado passar dali.

Mas às vezes nem sempre o Linux pode ter o que é necessário, por exemplo se for um utilizador que não dispense os jogos complexos o Linux não tem qualquer suporte para eles, poucos jogos desse tipo são feitos para Linux, outro exemplo são programas de cinema ou modelação e render onde requer grande poder gráfico e por norma constroem só para Windows e MAC.

Por outro lado o que o Linux tem, o Windows também pode ter, exemplo: OpenOffice (Alternativa ao Microsoft Windows), isto ao facto de muitas das aplicações serem construídas em C, e se forem bem estruturadas podem ser compiladas para Windows sem qualquer problema.