

## Introdução

Este documento tem como requisito esclarecer a diferença entre Tabela de partição e Tipos de Partições, explicando também os tipos de tabelas/partições mais usadas ultimamente e o que difere entre a DOS/MBR da GPT.

## Tabela de Partição

Uma tabela de partição é uma tabela mantida no disco que descreve as partições daquele disco e que será usada pelo Sistema Operacional, essa tabela contém algumas informações úteis sobre a partição, incluindo seu `tipo`, `inodes`, `tempo de montagem`, `quantidade de montagens` entre outras informações.

Dependendo da tabela de partição, você fica limitado há um certo número de partições que podem ser criadas, os tipos de tabelas mais comuns são: DOS e GPT, em ferramentas como `fdisk` você pode mudar a tabela do disco ou de uma partição.

## Como funciona a tabela

Um disco para funcionar de "modo normal" deve ter uma tabela criada, isso é obrigatório para que o Sistema Operacional reconheça os dados dentro dessa tabela, existem aplicações onde o disco não tem uma tabela, e apenas uma aplicação específica consegue ler os dados, para o Sistema Operacional, um disco sem tabela é um disco vazio, sem dados, esse é o modo como o S.O enxerga, porém, podem ter dados sim dentro do disco.

Essa tabela (primeira tabela do disco) seria uma tabela master e cada partição criada herda essa tabela, só que, você pode mudar a tabela de uma partição, exemplo:

João tem um disco de 1TB, a tabela master está como DOS e João cria duas partições nesse disco, por padrão, as tabelas das partições são herdadas da master, ou seja, são do tipo DOS, mas João pega uma dessas partições e converte para GPT, a tabela primária master é DOS, mas a tabela de uma partição pode ser ou não ser, não é recomendado fazer este tipo de ação, por boas práticas você deve usar uma única tabela de partição e não um tipo de tabela para cada partição, mas cada caso é um caso, fica aqui esse aviso.

Um passatempo divertido que você pode fazer é ver como tudo isso funciona na prática, fazer backup da tabela, apagar ela, criar uma nova, voltar a tabela recriando as partições como estavam antes e ver que tudo irá funcionar perfeitamente, apesar de não ter mais tabela, os dados estão lá, inclusive os dados das partições, isso tudo acarreta num bom aprendizado e entendimento do assunto.

## Tabela MBR

Os HDs precisam ser particionados por diversos motivos, e mesmo que não precisem, quando você formata um HD, é criada uma única partição (`sda1`) contendo todo o armazenamento do HD naquela partição, caso tenha duas partições (`sda1` e `sda2`) você fez o que chamamos de particionamento de HD, ou seja, você dividiu ele em partes.

Para que isso seja possível, é alocado uma tabela no início do disco e ela define onde no disco reside cada partição, essa tabela é conhecida como **MBR** (Master Boot Record) e tem como responsabilidade a função de inicializar o Sistema Operacional, isso porque a MBR inclui um pequeno programa que lê os registros do setor de inicialização da partição onde o Sistema está instalado e o setor de inicialização dessa partição também é lido e após isso o pequeno programa que está no setor de inicialização é chamado (bootloader do grub) que por sua vez inicializa o grub que por sua vez começa a inicializar o Sistema.

A MBR também conhece todas as partições do disco, isso é muito importante, pois, é a MBR que define quantas partições o disco pode ter, o tamanho do disco que ele consegue reconhecer com o S.O entre outras informações.

## Limite de partições

Ao atingir o limite de partições primárias é retornado um erro `To create more partitions, first replace a primary with an extended partition` (Para criar mais partições, primeiro substitua uma partição primária por uma partição estendida).

```
Disk /dev/sda3: 5.102 GiB, 6441402368 bytes, 12580864 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x02b0fc95
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda3p1		2048	2099199	2097152	1G	83	Linux
/dev/sda3p2		2099200	4196351	2097152	1G	83	Linux
/dev/sda3p3		4196352	6293503	2097152	1G	83	Linux
/dev/sda3p4		6293504	8390655	2097152	1G	83	Linux

A nomenclatura mudou porque criamos uma partição dentro de outra partição e só fizemos isso para que tivéssemos uma tabela de partição diferente da tabela master, mas novamente, isso não é recomendado, tenha apenas uma única tabela e suas partições, não crie tabelas dentro de tabelas isso é feio.

## Tabela GPT

Apesar do GPT substituir o MBR, o primeiro setor do disco fica reservado para uma pequena tabela MBR, denominada **Protective Master Boot Record**, sua única função é mera compatibilidade de software e não é usada, o processo de inicialização de um disco GPT é idêntico ao disco MBR, isso graças ao cabeçalho **Protective Master Boot Record** que é encontrado no primeiro setor do disco. Com um disco GPT, você pode ver essa informação usando o comando `gdisk` (Interactive GUID partition table (GPT) manipulator).

Ao rodar o 'gdisk -l device', ele exibe umas informações, inclusive um scan do dispositivo.

Partition table scan:

MBR: protective  
BSD: not present  
APM: not present  
GPT: present

Perceba que ele identificou a tabela de partição como GPT e a tabela MBR como protective, isso porque ocorre o processo explicado acima.

Além desse pequeno detalhe, um disco com uma tabela GPT contém 2 cópias do cabeçalho GPT, a primeira inicia no segundo setor do disco, logo após a MBR, a segunda cópia fica no ultimo setor do disco. Na verdade não temos apenas os cabeçalhos do GPT separados em 2 cópias, temos também os dados das tabelas, por esse motivo o GPT é mais seguro que a MBR, pois o GPT possui cópia de seus dados, já a MBR só possui essa informação em apenas um setor.

## Limite de partições

Uma partição foi convertida para GPT tendo uma tabela master como DOS, perceba que a nomenclatura das partições muda um pouco e perceba como posso ter mais de 4 partições primárias.

```
Disk /dev/sda1: 30 GiB, 32212254720 bytes, 62914560 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 83DF2397-5910-5D4F-A284-725A5343D9F5
```

Device	Start	End	Sectors	Size	Type
/dev/sda1p1	2048	2099199	2097152	1G	Linux filesystem
/dev/sda1p2	2099200	4196351	2097152	1G	Linux filesystem
/dev/sda1p3	4196352	6293503	2097152	1G	Linux filesystem
/dev/sda1p4	6293504	8390655	2097152	1G	Linux filesystem
/dev/sda1p5	8390656	10487807	2097152	1G	Linux filesystem
/dev/sda1p6	10487808	12584959	2097152	1G	Linux filesystem

## Diferenças entre as tabelas

### Master Boot Record (MBR/DOS)

Um disco pode dividir-se em até no máximo 4 partições primárias, possui um tamanho máximo de 2TB por disco e toda a informação das partições é guardada em apenas num único local, se a MBR for corrompida, você perde todo o disco (uma nova pode ser feita, mas terá a perda dos dados).

### GUID Partition Table (GPT)

Aqui temos um aumento significativo no limite de partições primárias, podendo ter até no máximo 128 partições primárias (contra 4 da tabela MRB/DOS), suporte para discos acima dos 2 TB, também temos mecanismos para detecção de dados e partições corrompidas entre outras melhorias.

## Tipo de Partição

O tipo de partição é armazenado dentro da tabela de partição criada e define o que a partição representa logicamente, é mais como um identificador daquela partição para nós humanos, o que vai definir mesmo a função daquela partição é o FS (File System ou Sistemas de arquivos) da partição. Fiel System é uma maneira de armazenar dados dentro das partições, de uma maneira que seja fácil de gerenciar, ler e gravar dados nela.

Por exemplo, uma partição do tipo `swap` diz aos humanos que essa partição será usada como troca (swap), mas ainda é preciso definir um FS (File System) que consiga gerenciar isso, por isso, uma partição do tipo `swap` recebe um FS `swap`, que vai gerenciar as trocas melhor que um FS do tipo `ext4`.

Uma partição do tipo Linux (código 83) recebe um FS que rode nesse tipo de partição, como o `ext4` por exemplo.

Atualmente, os tipos de partições mais usados no Linux são: `EXT4`, `BRTFS`, `XFS`, `SWAP`, `BIOS Boot` entre outros.