RAID
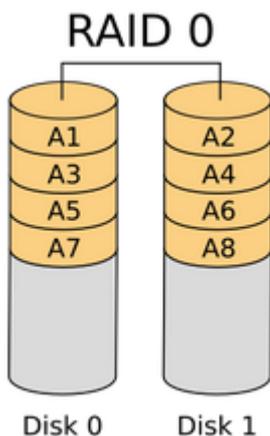
Illustration sources:

- https://en.wikipedia.org/wiki/Standard_RAID_levels
- https://en.wikipedia.org/wiki/Nested_RAID_levels

RAID (Redundant Array of Independent Disks) is a storage organisation technique that combines multiple hard drives into a single logical drive to increase performance, reliability and capacity. There are several standard RAID levels, each with its own unique features, benefits and limitations.
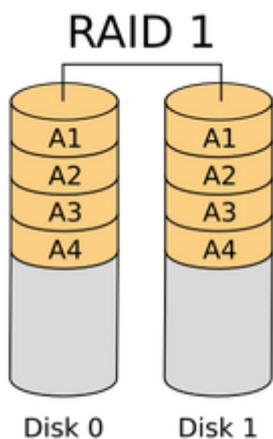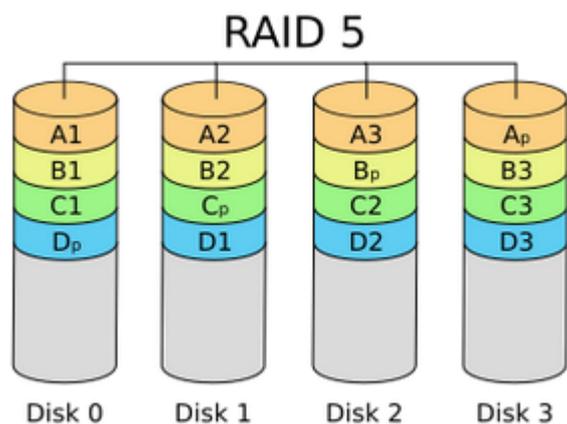
Types of RAID

RAID 0 - Striping

1. **Description**: RAID 0 divides data into blocks and writes to all drives in parallel, increasing transfer rates.
2. **Advantages**:
    1. High write and read performance.
    2. The entire available capacity of the disks is used.
3. **Disadvantages**:
    1. No redundancy; failure of one drive leads to loss of all data.
4. **Example**: Two drives of 1TB each in a RAID 0 configuration offer 2TB capacity, but no redundancy.
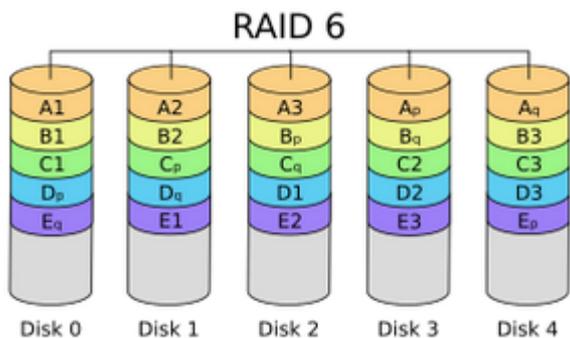
RAID 1 - Mirroring

RAID 1

Disk 0   Disk 1

1. **Description**: RAID 1 creates an exact copy of the data on at least two drives.
2. **Advantages**:
    1. High reliability; failure of one drive does not cause data loss.
    2. Fast readout, as data can be read from any of the disks.
3. **Disadvantages**:
    1. Capacity level equal to 50% of the total capacity of the drives in the array.
4. **Example**: Two drives of 1 TB each in RAID 1 provide 1 TB of capacity.
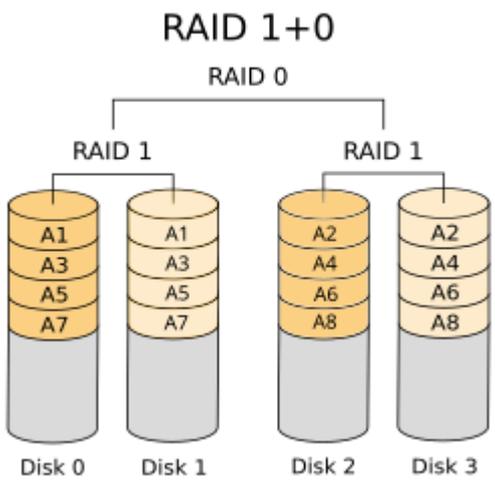
RAID 5 - Striping with parity

RAID 5

Disk 0   Disk 1   Disk 2   Disk 3

1. **Description**: RAID 5 combines striping with distributed parity, meaning that data and parity are stored on different drives.
2. **Advantages**:
    1. High performance and increased fault tolerance (possibility of losing one drive).
3. **Disadvantages**:
    1. Requires at least three disks.
    2. Complexity of parity calculation can affect write performance.
4. **Example**: Three disks of 1 TB each in RAID 5 provide 2 TB of capacity (1 TB per parity).

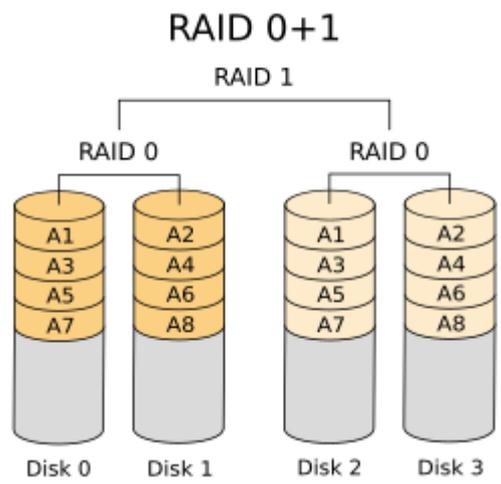## RAID 6 - Striping with double parity

RAID 6

Disk 0   Disk 1   Disk 2   Disk 3   Disk 4

1. **Description**: RAID 6 works similarly to RAID 5, but with an additional parity block, allowing two drives to fail.
2. **Advantages**:
    1. Even higher reliability than RAID 5.
3. **Disadvantages**:
    1. Requires a minimum of four drives.
    2. Lower write performance compared to RAID 5 due to additional parity.
4. **Example**: Four drives of 1 TB each in RAID 6 offer 2 TB of capacity.

RAID 10 - Striping and mirroring

RAID 1+0

1. **Description**: RAID 10 combines striping (RAID 0) and mirroring (RAID 1) for both high performance and data security.
2. **Advantages**:
    1. High performance and redundancy.
3. **Disadvantages**:
    1. Requires at least four drives.
    2. Only 50% of total capacity is available as usable capacity.
4. **Example**: Four drives of 1 TB each in RAID 10 provide 2 TB of capacity.

RAID 01 - Mirroring and striping

Description:

RAID 01 (i.e. RAID 0+1) is a combination of RAID 0 and RAID 1. A copy (mirror) of two or more groups of disks is created, which is then divided into striping.

Advantages:

Offers better performance than pure RAID 1 and redundancy. Allows data recovery in the event of failure of one of the drives in each group.

Disadvantages:

Requires at least four drives. If two disks in the same group fail, data may be lost. Capacity level equal to 50% of total disk capacity.

RAID capacity calculation

The capacity of a RAID array can be calculated in different ways, depending on the RAID level selected. Here are the general rules:

1. **RAID 0**: Capacity = (Number of drives) x (Capacity of one drive).
2. **RAID 1**: Capacity = (Capacity of one disk) x (Number of disks / 2).
3. **RAID 5**: Capacity = (Number of disks - 1) x (Capacity of one disk).
4. **RAID 6**: Capacity = (Number of disks - 2) x (Capacity of one disk).
5. **RAID 10**: Capacity = (Capacity of one disk) x (Number of disks / 2).
6. **RAID 01**: Capacity = (Capacity of one disk) x (Number of disks / 2).

Calculation example

Suppose we have 4 disks of 2 TB each:

1. **RAID 0**: 4 x 2 TB = 8 TB
2. **RAID 1**: 2 TB x 2 = 2 TB
3. **RAID 5**: (4 - 1) x 2 TB = 6 TB
4. **RAID 6**: (4 - 2) x 2 TB = 4 TB
5. **RAID 10**: 2 TB x 2 = 4 TB
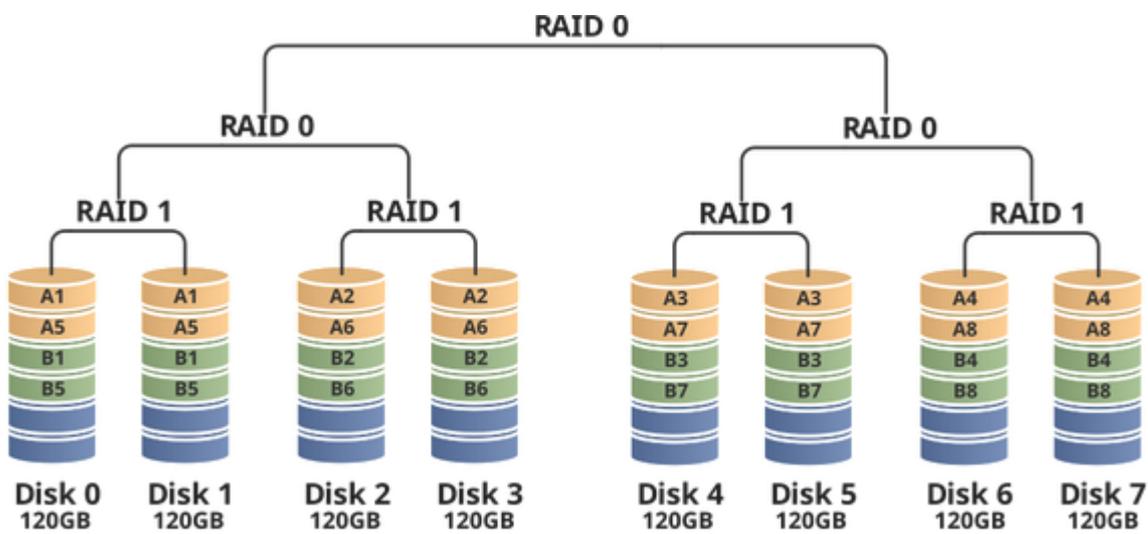
6. **RAID 01**: 2 TB x 2 = 4 TB

Summary

RAID configuration allows you to optimise the performance levels, reliability and capacity of your data storage systems. Understanding the types of RAID and how their capacities are calculated is key to choosing the right configuration for the tasks you want to perform, and to ensure that your data is protected from loss.

Curiosity

What will this RAID be called and what capacity will it have?

RAID calculator

code:

```html
<!DOCTYPE html>
<html lang="pl">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>RAID Capacity Calculator</title>
    <style>
        body {
            background-color: white;
            font-family: Arial, sans-serif;
            text-align: center;
            margin-top: 50px;
        }
        form {
            display: inline-block;
```

```
            padding: 20px;
            border: 1px solid #ccc;
            border-radius: 5px;
        }
    </style>
</head>
<body>

<h1>RAID Capacity Calculator</h1>

<form method="post" action="">
    <label for="numDisks">Ilość dysków:</label>
    <input type="number" name="numDisks" id="numDisks" min="1"
required><br><br>

    <label for="diskSize">Pojemność dysku (w TB):</label>
    <input type="number" name="diskSize" id="diskSize" min="1"
required><br><br>

    <label for="raidType">Typ RAID:</label>
    <select name="raidType" id="raidType">
        <option value="0">RAID 0</option>
        <option value="1">RAID 1</option>
        <option value="5">RAID 5</option>
        <option value="6">RAID 6</option>
        <option value="10">RAID 10</option>
    </select><br><br>

    <input type="submit" name="calculate" value="Oblicz pojemność">
</form>

<?php
if (isset($_POST['calculate'])) {
    $numDisks = (int)$_POST['numDisks'];
    $diskSize = (float)$_POST['diskSize'];
    $raidType = (int)$_POST['raidType'];

    $capacity = 0;

    switch ($raidType) {
        case 0: // RAID 0
            $capacity = $numDisks * $diskSize;
            break;
        case 1: // RAID 1
            $capacity = ($numDisks / 2) * $diskSize;
            break;
        case 5: // RAID 5
            $capacity = ($numDisks - 1) * $diskSize;
            break;
        case 6: // RAID 6
            $capacity = ($numDisks - 2) * $diskSize;
```

```php
                break;
        case 10: // RAID 10
            $capacity = ($numDisks / 2) * $diskSize;
            break;
        default:
            $capacity = 0;
            break;
    }

    echo "<h2>Całkowita pojemność RAID: $capacity TB</h2>";
}
?>


</body>
</html>
```