Apache Spark Memory Management

# Overview of Spark Memory Management

Apache Spark's memory management is crucial for optimizing performance and resource utilization. Understanding how Spark allocates and manages memory can help in tuning applications effectively.

# Types of Memory in Spark

## 1. Reserved Memory

Purpose: Reserved for Spark's internal operations and metadata.  
Default Size: 300 MB.  
Note: This memory is not configurable and is subtracted from the total JVM heap memory.

## 2. User Memory

Purpose: Used for user-defined data structures and variables.  
Allocation: Accounts for the remaining memory after allocating Spark memory (execution and storage).

## 3. Spark Memory (Unified Memory)

Purpose: Shared between execution and storage tasks.  
Controlled By:  
- spark.memory.fraction (default: 0.6): Fraction of (JVM heap - reserved memory) allocated to Spark memory.  
- spark.memory.storageFraction (default: 0.5): Fraction of Spark memory allocated to storage.

### a. Execution Memory

Used For: Computations like shuffles, joins, sorts, and aggregations.  
Behavior: If execution memory needs more space, it can evict storage memory.

### b. Storage Memory

Used For: Caching RDDs, DataFrames, and broadcast variables.  
Behavior: If storage memory needs more space, it cannot evict execution memory.

## 4. Off-Heap Memory

Purpose: Memory allocated outside the JVM heap to reduce garbage collection overhead.  
Controlled By:  
- spark.memory.offHeap.enabled (default: false)  
- spark.memory.offHeap.size (default: 0)

# Memory Layout Diagram

Below is a simplified diagram illustrating Spark’s memory layout

