**Efficient Inventory Management System**

**Core Challenges**

Managing an inventory system effectively involves handling large volumes of data and ensuring that operations such as accessing, updating, and removing product information are performed swiftly. This requires selecting the right data structures and algorithms.

**Optimal Data Structures**

1. **HashMap**: This data structure provides fast access to products using unique keys. It is well-suited for operations like insertion, lookup, and deletion, offering constant-time performance on average.
2. **ArrayList**: This structure is beneficial for maintaining a dynamic list of products with indexed access. While it supports efficient indexing, it is generally slower for lookups compared to a HashMap.
3. **LinkedList**: This structure is ideal for scenarios with frequent insertions and deletions but is less efficient for lookups compared to a HashMap. It is best used when the primary operations involve modifying the list.

**Performance Metrics**

1. **Adding a Product**:
   * **Operation**: addProduct
   * **Complexity**: O(1) on average. The HashMap’s constant-time performance ensures efficient product insertion.
2. **Updating a Product**:
   * **Operation**: updateProduct
   * **Complexity**: O(1) on average. HashMap allows for constant-time updates, making modifications quick and efficient.
3. **Removing a Product**:
   * **Operation**: deleteProduct
   * **Complexity**: O(1) on average. HashMap supports constant-time deletions, facilitating rapid removal of products.

**Enhancing Efficiency**

To further optimize the performance of the inventory system, consider the following strategies:

* **Load Factor Optimization**: Adjust the load factor of the HashMap to balance performance between time and space complexity, ensuring the system operates efficiently.
* **Concurrency Handling**: Use ConcurrentHashMap if the system needs to support multiple users accessing and modifying data simultaneously. This ensures thread safety and reduces conflicts.
* **Database Integration**: For handling complex queries and larger datasets, integrating a database with indexing capabilities can improve search and retrieval performance.

**Role of Data Structures and Algorithms**

Effective inventory management relies on choosing the appropriate data structures and implementing efficient algorithms. Data structures help organize and store data efficiently, while algorithms provide the methods for manipulating this data, aiming to minimize both time and space complexity.

By employing these techniques and leveraging efficient data structures such as HashMap, you can achieve a well-optimized inventory management system with minimized time complexity for core operations.