LinkedTransferQueue class

Java TransferQueue is a concurrent blocking queue implementation in which producers may wait for receipt of messages by consumers. LinkedTransferQueue class is an implementation of `TransferQueue` in Java.

TransferQueue may be useful for example in message passing applications in which producers sometimes (using method transfer()) await receipt of elements by consumers invoking take or poll, while at other times enqueue elements (via method put()) without waiting for receipt.

When a producer reaches to TransferQueue to transfer a message and there are consumers waiting to take message, then producer directly transfers the message to consumer.

If there is no consumer waiting, then producer will not directly put the message and returned, rather it will wait for any consumer to be available to consume the message.

1. LinkedTransferQueue Features

Let’s note down few important points on the LinkedTransferQueue in Java.

- LinkedTransferQueue is an unbounded queue on linked nodes.
- This queue orders elements FIFO (first-in-first-out) with respect to any given producer.
- Elements are inserted at the tail, and retrieved from the head of the queue.
- It supplies blocking insertion and retrieval operations.
- It does not allow NULL objects.
- LinkedTransferQueue is thread safe.

- The size() method is NOT a constant-time operation because of the asynchronous nature, so may report inaccurate results if this collection is modified during traversal.
- The bulk operations addAll, removeAll, retainAll, containsAll, equals, and toArray are not guaranteed to be performed atomically. For example, an iterator operating concurrently with an addAll operation might view only some of the added elements.