1. Understanding Hadoop-specific Data Types

\* How to configure input and output data types.

2. Custom Data Types

\* Implementing a Custom Hadoop Writable Data Type

\*Implementing a Custom Hadoop Key Type

3. Understanding InputFormats

\*Implementing a Custom Input Format

4. Understanding SequenceFiles

5. Hadoop Counters for reporting Custom Metrics.

6. Testing Hadoop programs with MRUnit.

7. Performing Joins on Data

\*Map-side versus reduce-side joins

8. Using the Distributed Cache

Sequence Files

===========

The SequenceFile class within the org.apache.hadoop.io package provides an efficient binary file format that is often useful as an output from a MapReduce job. This is especially true if the output from the job is processed as the input of another job. The Sequence files have several advantages, as follows:

. As binary files, they are intrinsically more compact than text files

. They additionally support optional compression, which can also be applied at different levels, that is, compress each record or an entire split

. The file can be split and processed in parallel