

```
DataOutputStream Write Time: 12497 ms
OutputStreamWriter Write Time: 1737 ms
DataInputStream Read Time: 8338 ms
InputStreamReader Read Time: 1146 ms
```

当使用 `DataOutputStream` 和 `OutputStreamWriter` 写入大量的 `double` 数据，然后使用 `DataInputStream` 和 `InputStreamReader` 读取这些数据时，您会注意到时间上的差异。这些差异主要由以下几个因素导致：

#### 写入过程

##### `DataOutputStream`:

**二进制格式：**`DataOutputStream` 以二进制格式写入数据。对于 `double` 数据，它使用固定的 8 字节来表示每个数值。

**效率：**由于直接写入二进制数据，不需要转换为字符串，所以这个过程通常更快、更高效。

**文件大小：**生成的文件相对较小，因为每个 `double` 数据占用固定的字节大小。

##### `OutputStreamWriter`:

**文本格式：**`OutputStreamWriter` 将数据转换为字符形式（字符串），然后写入文件。这意味着 `double` 数据会被转换为它们的字符串表示形式。

**效率：**将 `double` 数据转换为字符串需要额外的处理时间，因此写入过程可能较慢。

**文件大小：**生成的文件可能更大，因为 `double` 数据的字符串表示通常会占用多于 8 字节的空间。

读取过程

DataInputStream:

二进制格式：以二进制形式直接读取数据，不需要进行转换。

效率：通常读取速度更快，因为它直接读取固定大小的二进制数值。

InputStreamReader:

文本格式：读取的是文本数据，需要将字符串解析转换回 double 数据。

效率：解析字符串到 double 的过程需要额外的时间，因此读取过程可能较慢。

错误处理：文本格式的数据更容易受到格式错误或编码问题的影响，这可能导致额外的错误检查和处理。

结论

二进制 vs. 文本：DataOutputStream 和 DataInputStream 以二进制格式处理数据，通常比处理文本格式（如 OutputStreamWriter 和 InputStreamReader）更快。

转换开销：将 double 数据转换为字符串（和反向转换）需要额外的时间和资源。

文件大小和 IO 开销：二进制格式的文件通常较小，这意味着磁盘 I/O 操作可能更快。

在实际应用中，选择哪种方式取决于具体需求：如果对性能有较高要求，二进制格式通常是更好的选择。但如果需要易于阅读和处理的数据格式（例如文本），则可能会选择 OutputStreamWriter 和 InputStreamReader。