

Item 194

**git\_comments:**

**git\_commits:**

1. **summary:** [FLINK-13632] Remove old PojoSerializerSnapshotMigrationTest  
**message:** [FLINK-13632] Remove old PojoSerializerSnapshotMigrationTest

**github\_issues:**

**github\_issues\_comments:**

**github\_pulls:**

**github\_pulls\_comments:**

**github\_pulls\_reviews:**

**jira\_issues:**

1. **summary:** Port serializer upgrade tests to TypeSerializerUpgradeTestBase  
**description:** FLINK-11767 introduced a new test base (`{{TypeSerializerUpgradeTestBase}}`) for writing serializer upgrade tests. Now we need to migrate all existing tests from `{{TypeSerializerSnapshotMigrationTestBase}}` to use the new test base and update them to restore from Flink 1.8 onward. It seems these are the subclasses: `* TtlSerializerStateMigrationTest (org.apache.flink.runtime.state.ttl) * ValueSerializerMigrationTest (org.apache.flink.api.java.typeutils.runtime) * PrimitiveArraySerializerSnapshotMigrationTest (org.apache.flink.api.common.typeutils.base.array) * AvroSerializerMigrationTest (org.apache.flink.formats.avro.typeutils) * TupleSerializerMigrationTest (org.apache.flink.api.java.typeutils.runtime) * BufferEntrySerializerMigrationTest (org.apache.flink.streaming.api.operators.co) * TimerSerializerSnapshotMigrationTest (org.apache.flink.streaming.api.operators) * StreamElementSerializerMigrationTest (org.apache.flink.streaming.runtime.streamrecord) * KryoSnapshotMigrationTest (org.apache.flink.api.java.typeutils.runtime.kryo) * BaseTypeSerializerSnapshotMigrationTest (org.apache.flink.api.common.typeutils.base) * NullableSerializerMigrationTest (org.apache.flink.api.java.typeutils.runtime) * VoidNamespacieSerializerSnapshotMigrationTest (org.apache.flink.runtime.state) * ScalaOptionSerializerSnapshotMigrationTest (org.apache.flink.api.scala.typeutils) * ScalaTrySerializerSnapshotMigrationTest (org.apache.flink.api.scala.typeutils) * JavaSerializerSnapshotMigrationTest (org.apache.flink.runtime.state) * LockableTypeSerializerSnapshotMigrationTest (org.apache.flink.cep.nfa.sharedbuffer) * NFASerializerSnapshotsMigrationTest (org.apache.flink.cep) * WindowSerializerSnapshotsMigrationTest (org.apache.flink.streaming.runtime.operators.windowing) * UnionSerializerMigrationTest (org.apache.flink.streaming.api.datastream) * TwoPhaseCommitSinkStateSerializerMigrationTest (org.apache.flink.streaming.api.functions.sink) * KafkaSerializerSnapshotsMigrationTest (org.apache.flink.streaming.connectors.kafka) * KafkaSerializerSnapshotsMigrationTest (org.apache.flink.streaming.connectors.kafka) * RowSerializerMigrationTest (org.apache.flink.api.java.typeutils.runtime) * ValueArraySerializerSnapshotMigrationTest (org.apache.flink.graph.types.valuearray) * MapSerializerSnapshotMigrationTest (org.apache.flink.api.common.typeutils.base) * CompositeTypeSerializerSnapshotMigrationTest (org.apache.flink.api.common.typeutils) * ListSerializerSnapshotMigrationTest (org.apache.flink.api.common.typeutils.base) * EnumSerializerSnapshotMigrationTest (org.apache.flink.api.common.typeutils.base) * PojoSerializerSnapshotMigrationTest (org.apache.flink.api.java.typeutils.runtime) * ArrayListSerializerMigrationTest (org.apache.flink.runtime.state) * CopyableSerializerMigrationTest (org.apache.flink.api.java.typeutils.runtime) * WritableSerializerMigrationTest (org.apache.flink.api.java.typeutils.runtime) * ListViewSerializerSnapshotMigrationTest (org.apache.flink.table.dataview) * MapViewSerializerSnapshotMigrationTest (org.apache.flink.table.dataview) * ScalaEitherSerializerSnapshotMigrationTest (org.apache.flink.api.scala.typeutils) * LongValueWithProperHashCodeSerializerSnapshotMigrationTest`

```
(org.apache.flink.graph.drivers.transform) * ScalaCaseClassSerializerSnapshotMigrationTest
(org.apache.flink.api.scala.typeutils) * TraversableSerializerSnapshotMigrationTest
(org.apache.flink.api.scala.typeutils) * EnumValueSerializerSnapshotMigrationTest
(org.apache.flink.api.scala.typeutils)
```

#### jira\_issues\_comments:

1. I'd say FLINK-15474 soft blocks this because doing the migration "the old way", by adding binary snapshots, is not quite maintainable, IMO.
2. We can start porting independently of FLINK-15474 because that change is almost transparent to the serializer test implementation, as can be seen in the linked PoC.
3. By the way, I think this is more of a "soft blocker", we didn't have these upgrade tests for a couple releases now and nothing seems to have broken. It would still be good to have them at some point, though.
4. I'm currently porting {{AvroSerializerMigrationTest}}.
5. I'm currently porting {{ValueSerializerMigrationTest}}
6. I'm porting {{TtlSerializerStateMigrationTest}}.
7. I'm porting {{PrimitiveArraySerializerSnapshotMigrationTest}}.
8. I'm on {{TupleSerializerMigrationTest}} now.
9. I'm on {{BufferEntrySerializerMigrationTest}} now.
10. I'm on {{TimerSerializerSnapshotMigrationTest}} now.
11. I'm on {{StreamElementSerializerMigrationTest}} now.
12. **body:** I could go faster but this is so boring that I can only do a few at a time. 🤖  
**label:** code-design
13. I'm on {{KryoSnapshotMigrationTest}}
14. I'm on {{BaseTypeSerializerSnapshotMigrationTest}}.
15. I'm on {{NullableSerializerMigrationTest}}, and {{VoidNamespacieSerializerSnapshotMigrationTest}}.
16. on {{ScalaOptionSerializerSnapshotMigrationTest}}.
17. Implemented on release-1.11 in 4248de26c3500f86596c2d06c91a9c76c576f3cd and previous commits.
18. Implemented on master in aedb4068408cfcad6f258526b00fcbff7f40fb82 and previous commits.
19. For future reference, I think we should not update the snapshots for every release but only update them when we touch the serializers.