

Item 44

**git\_comments:**

1. **comment:** Order is important. Streaming views almost all use Combine internally. TODO (BEAM-4118) Remove this check once combiner lifting is implemented for the FnAPI.  
**label:** code-design

**git\_commits:**

1. **summary:** [BEAM-3513] Removing PrimitiveCombineGroupedValues override w/ FnAPI.  
**message:** [BEAM-3513] Removing PrimitiveCombineGroupedValues override w/ FnAPI. When using the FnAPI, make it so that Combine.GroupedValues are not replaced by the Dataflow specific PTransforms. The reason for this is to allow the Combines to be translated to a Dataflow pipeline as the constituent parts of the composite transform (in other words, send Dataflow a GBK and a ParDo, instead of a CombineGroupedValues black box). This does not fully accomplish the final goal of portable combines since this method doesn't do anything to help Combiner lifting, but it works for the case of the basic Combine that has been implemented. It successfully treats the Combine as a GBK and ParDo when not lifted. Additional changes will need to be made to detect when we want to lift Combines and in those situations handle that properly, which may mean putting the Combines in the black box.  
**label:** code-design

**github\_issues:**

**github\_issues\_comments:**

**github\_pulls:**

**github\_pulls\_comments:**

**github\_pulls\_reviews:**

**jira\_issues:**

1. **summary:** Use portable CombinePayload in Java DataflowRunner  
**description:** The Java-specific blobs transmitted to Dataflow need more context, in the form of portability framework protos.

**jira\_issues\_comments:**

1. Specifically <https://github.com/apache/beam/blob/master/runners/google-cloud-dataflow-java/src/main/java/org/apache/beam/runners/dataflow/DataflowPipelineTranslator.java#L772>
2. The Java Dataflow runner now uses the portable CombinePayload. The details can be seen in the PR that implemented this: <https://github.com/apache/beam/pull/5974>