Item 360
**git_comments:**

1. * * Return the number of windows before the next checkpoint including the current window. * @return Number of windows from checkpoint, 1 if the checkpoint will be after the current window
2. this is write once variable
3. gn.connectInputPort("ip1", reservoir1); gn.connectInputPort("ip2", reservoir2);
4. DefaultReservoir reservoir1 = new DefaultReservoir("ip1Res", 1024); DefaultReservoir reservoir2 = new DefaultReservoir("ip2Res", 1024);
5. **comment:** Adding some extra time for the windows to finish
   **label:** code-design
6. GenericOperator go = new GenericOperator();

**git_commits:**

1. **summary:** APEXCORE-360 #resolve Providing a way for operator to check how many windows till checkpoint.
   **message:** APEXCORE-360 #resolve Providing a way for operator to check how many windows till checkpoint.

**github_issues:**

**github_issues_comments:**

**github_pulls:**

**github_pulls_comments:**

**github_pulls_reviews:**

**jira_issues:**

1. **summary:** Add capability in the engine to let the operator know about the next checkpoint window
   **description:** Carryover from the older JIRA system.. The operator should be able to call an engine API to find out the number of windows till the next checkpoint. Implementation of iteration feature requires the need to know the number of windows from checkpoint. If the delay is 1 the delay operator needs to know about the window before checkpoint by beginWindow of that window so it can save the tuples in that window for replay in case of recovery in future. If the delay is 2 then it needs to know 2 windows before checkpoint. The idea is to compute the windows from checkpoint and make it available in OperatorContext, for example by a method getWindowsFromCheckpoint. The operator can check the windows from checkpoint in beginWindow. In the iteration case it can save the tuples accordingly.

**jira_issues_comments:**

1. **body:** This has been implemented and merged
   **label:** requirement