Sample for Building a DataStage Job Using Change Data Capture 1.5 Accessing ASIQ 12.4.0

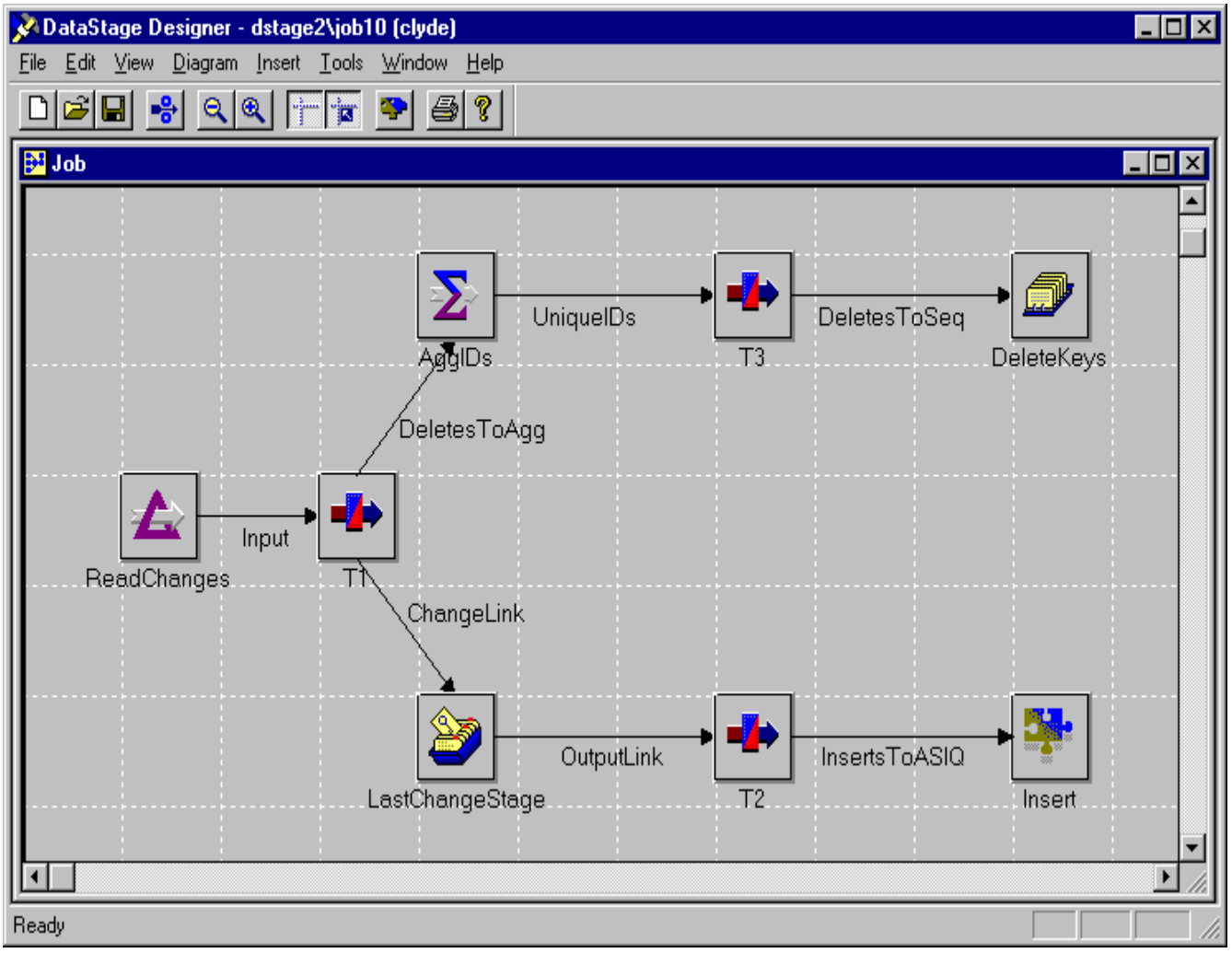
This sample provides a description of the DataStage Job used in the certification process to capture the latest replicated transactions from Oracle into ASIQ using CDC with Replication Server and RAO.

The steps to create this process follow and assume the following:

- All products are installed and configured properly (Ardent DataStage and CDC, ASIQ, ODBC, ASE, RAO, Repserver).
- The replication environment was configured and in production already.
- ODBC connectivity necessary for DataStage to communicate with ASIQ server is configured.
- The tables exist in ASIQ. In this particular case we are using a table with multiple datatypes.

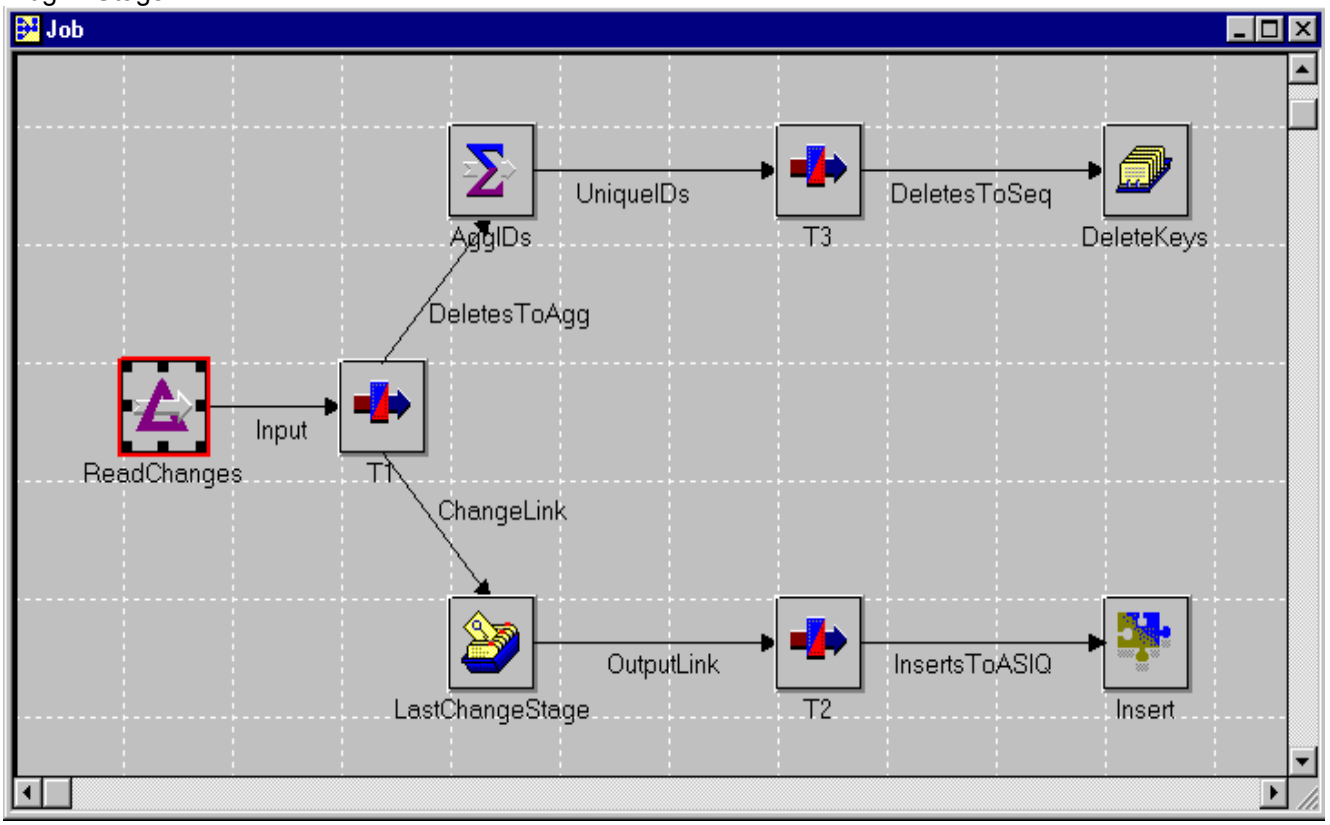
The following screen displays the DataStage Job that handles the logic of replicated events (INSERT, DELETE, UPDATE) coming from Oracle through Replication Server and CDC to pass into ASIQ. Remember that ASIQ servers do not support UPDATE statements. These transactions will be handled as a DELETE/INSERT transaction. A brief description of this Job could be as follows:

1. The replicated transactions coming from Repserver (Replication Server) and captured by CDC (Change Data Capture) are read by the DataStage CDC Plug-in.
2. Then the goal is to create two different files: one sequential file (DeleteKeys) that will hold all the DELETE and UPDATE transactions and a Hashed File (LastChangeStage) that will hold the last transaction for each replicated row.
3. Having these two files, load the Sequential file into a temporal table within ASIQ and define the right logic process (inside the ASIQ Plug-in Stage) to first delete the rows from the target table that are part of that Sequential file.
4. Load the data stored in the Hashed file passing only the INSERT and UPDATE transactions to ASIQ.

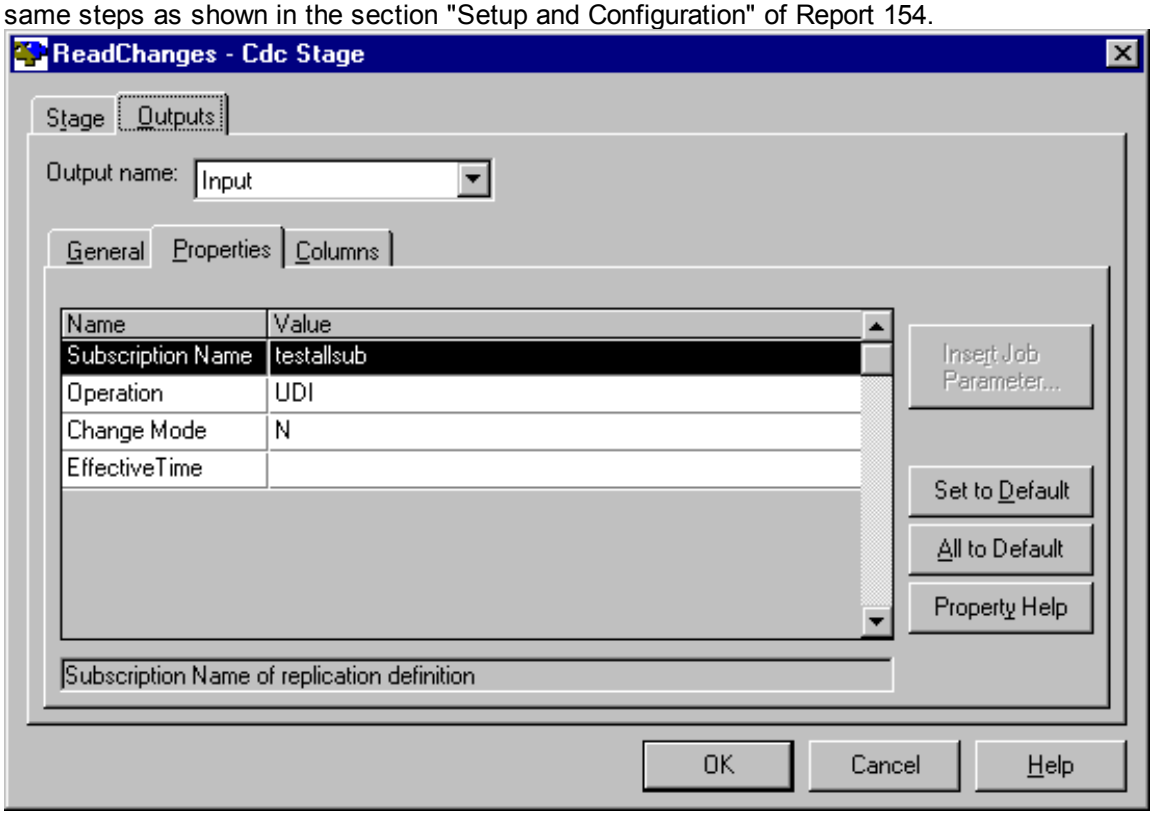


1. CDC Plug-in Stage (ReadChanges)

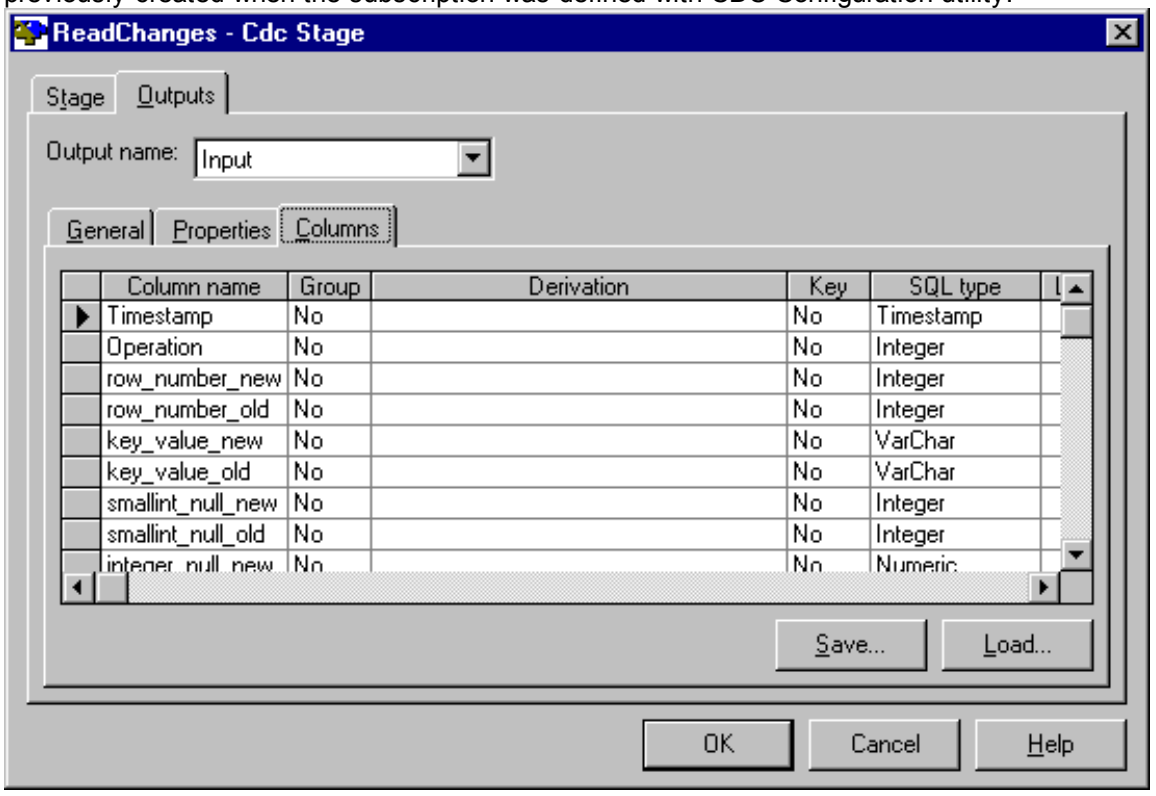
In the following sections, the most important description for each module will be discussed. The stage for each module will be marked as shown in the following screen. We will start the description of CDC Plug-in Stage.



1a) In the CDC Plug-in module, the following panels show the information required. In the **General** panel a project name is required. Within **Properties**, we will use the subscription definition that was defined using the same steps as shown in the section "Setup and Configuration" of Report 154.

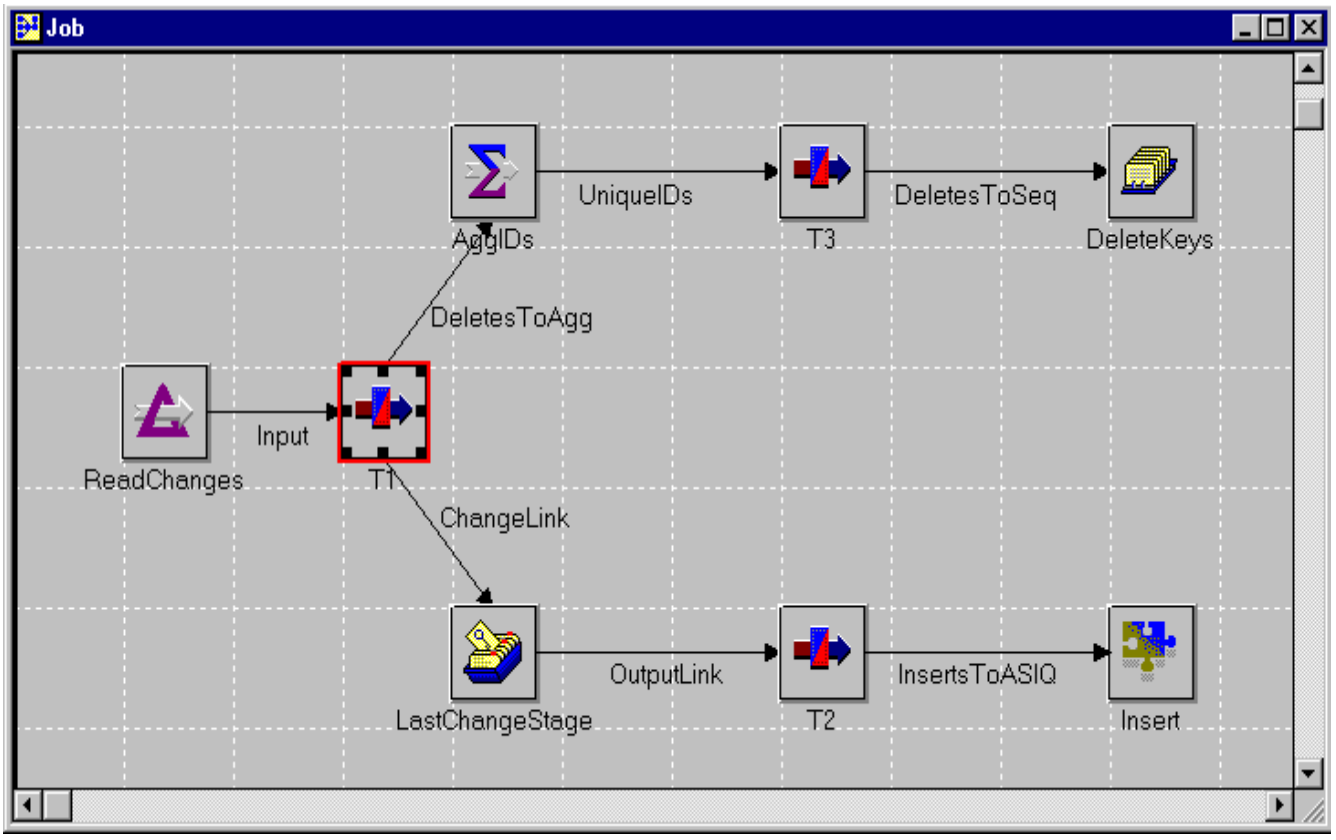


1b) The option Load (shown in the following screen) is used to get the column information, which was previously created when the subscription was defined with CDC Configuration utility.

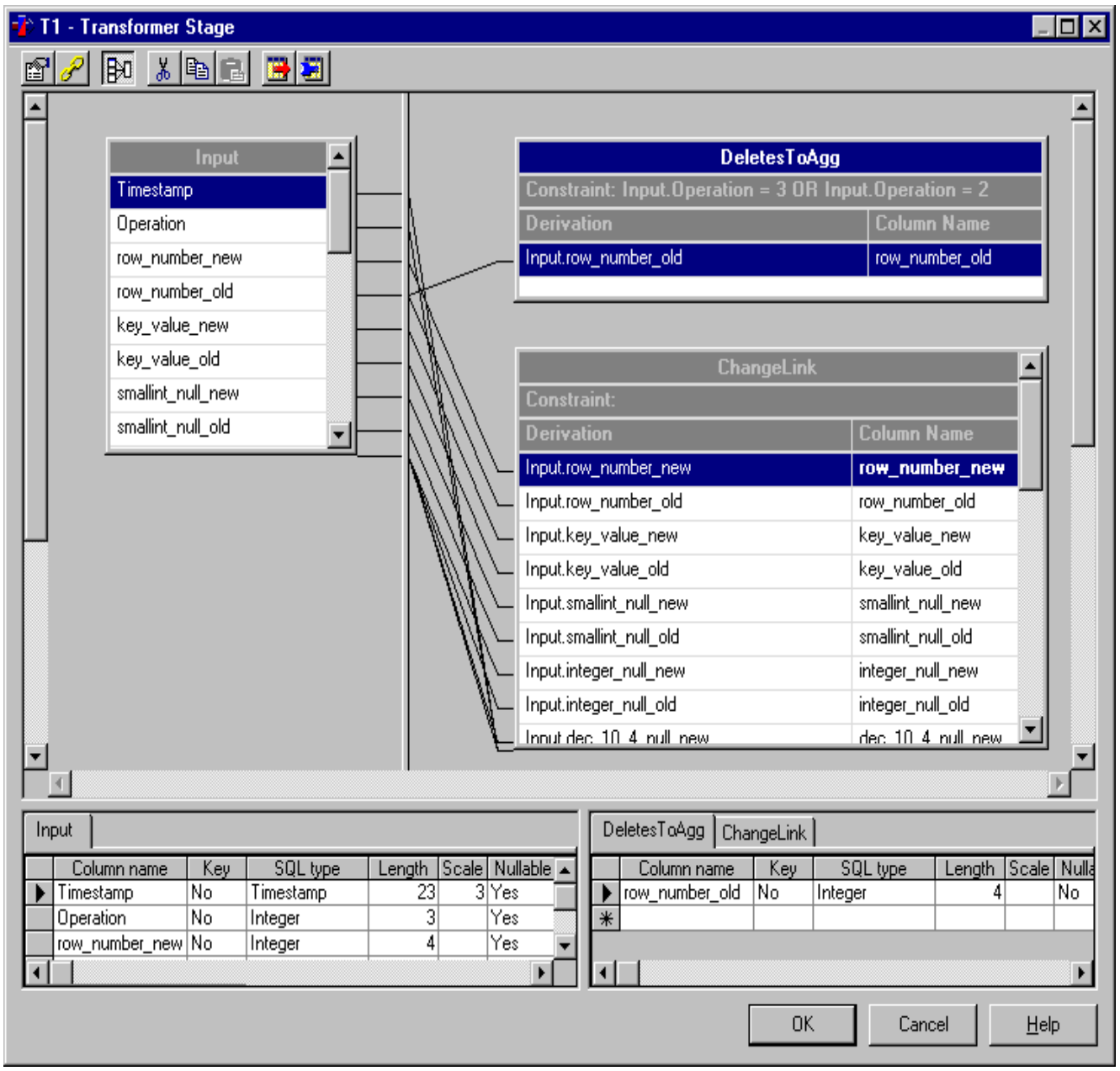


2.Transformation Stage (T1)

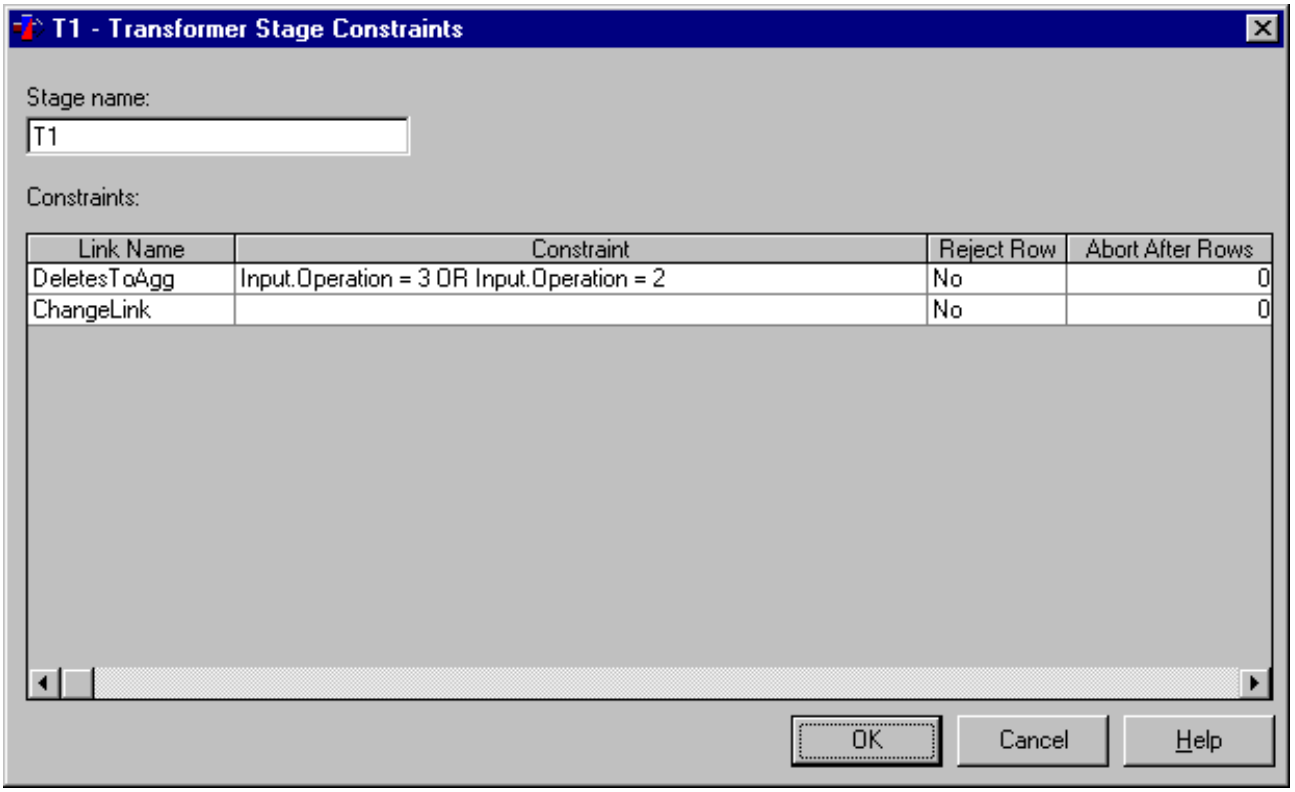
In this stage we will split the transactions depending on type (UPDATE, DELETES or INSERT) for further processing.



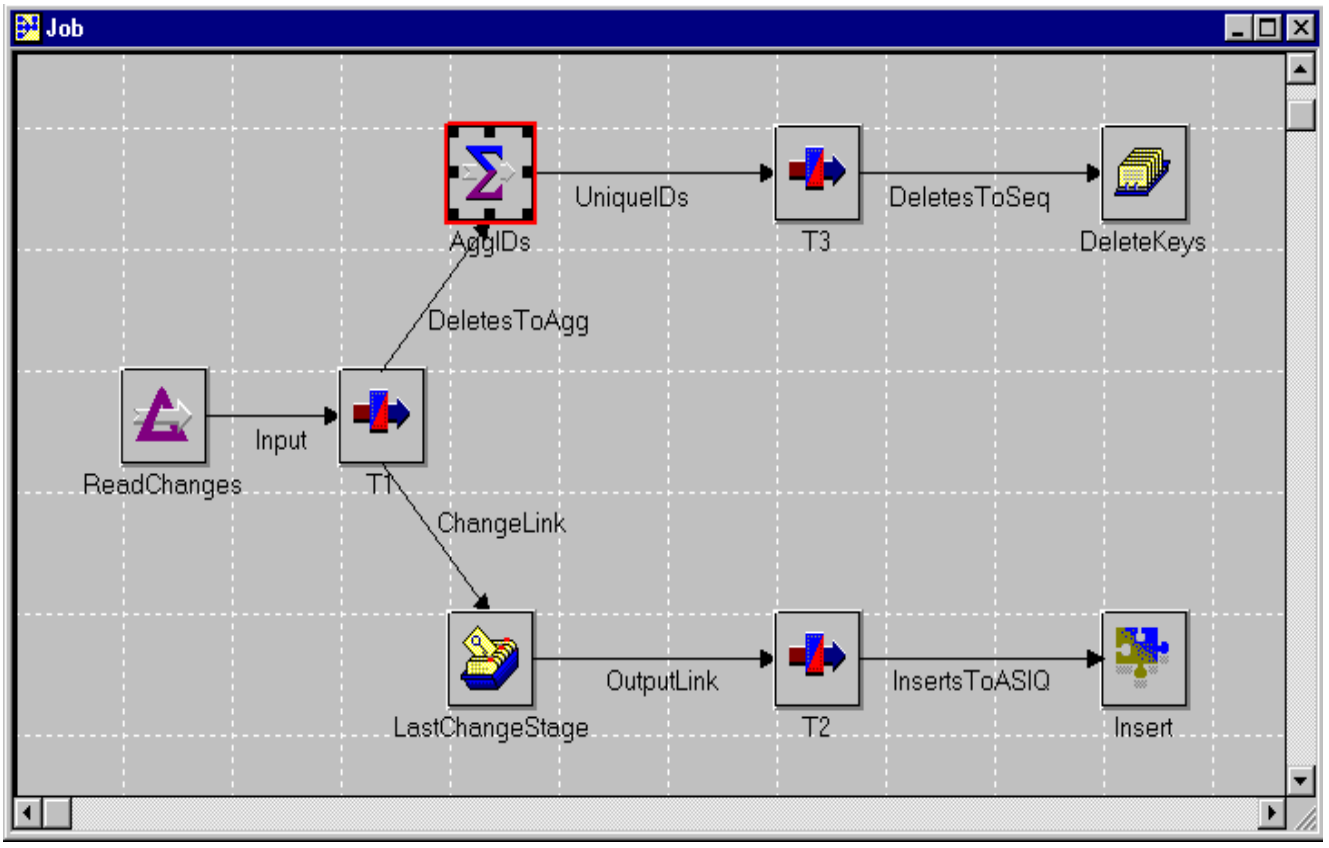
2a) Create the links between modules, as shown in the following screen.



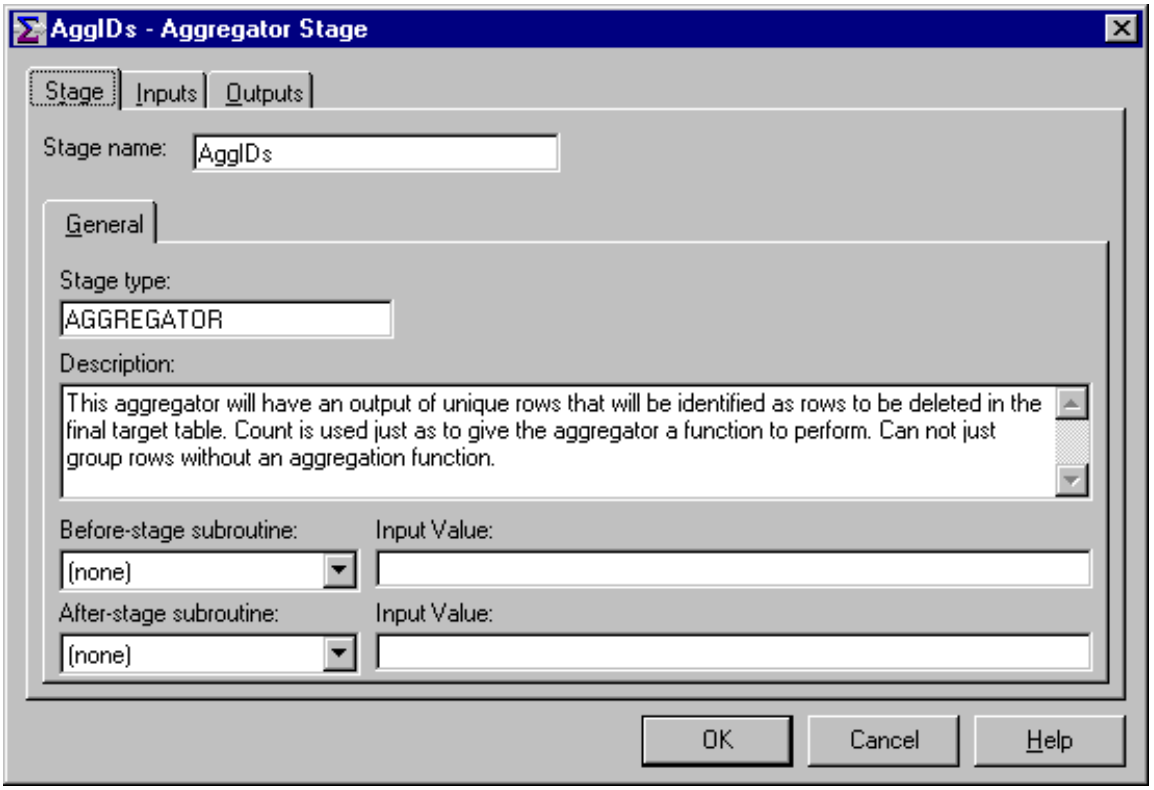
2b) Double click in the Constraint area to add it. In this case any operation marked as 2 (UPDATE) and 3 (DELETE) will be passed to link **DeletesToAgg**.



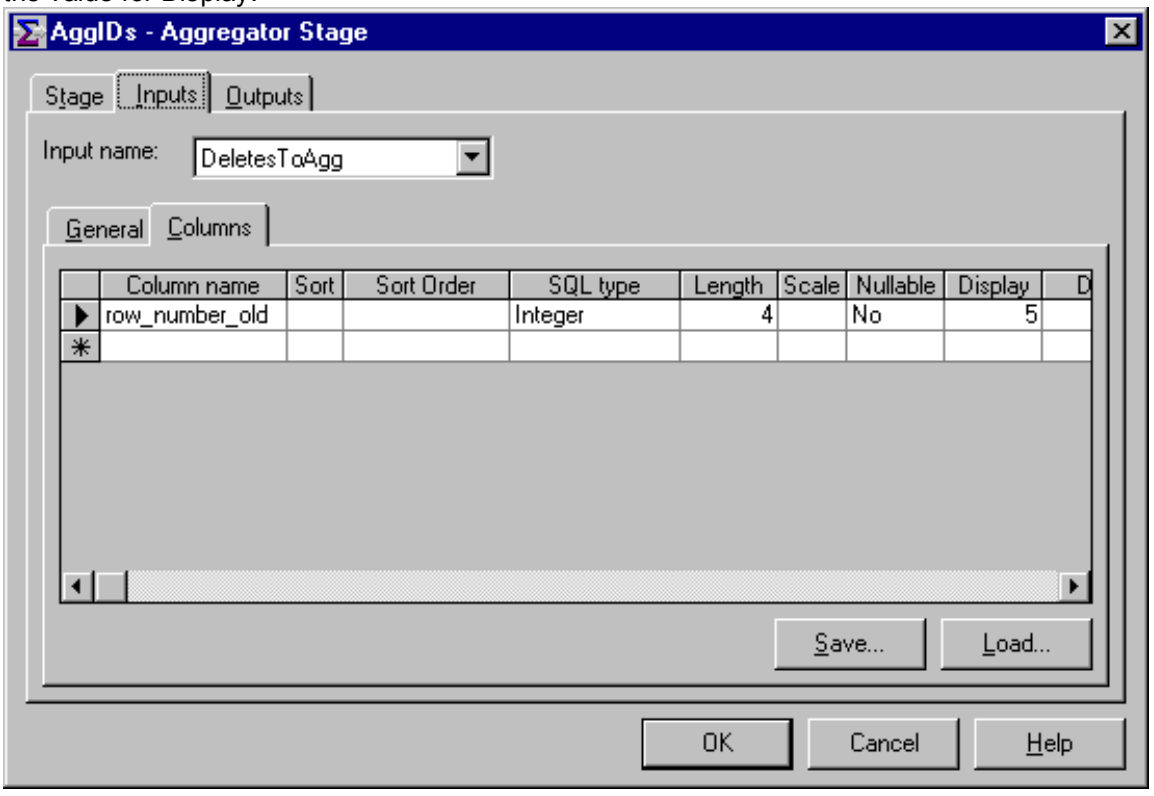
3.Aggregation Stage (AggIDs)



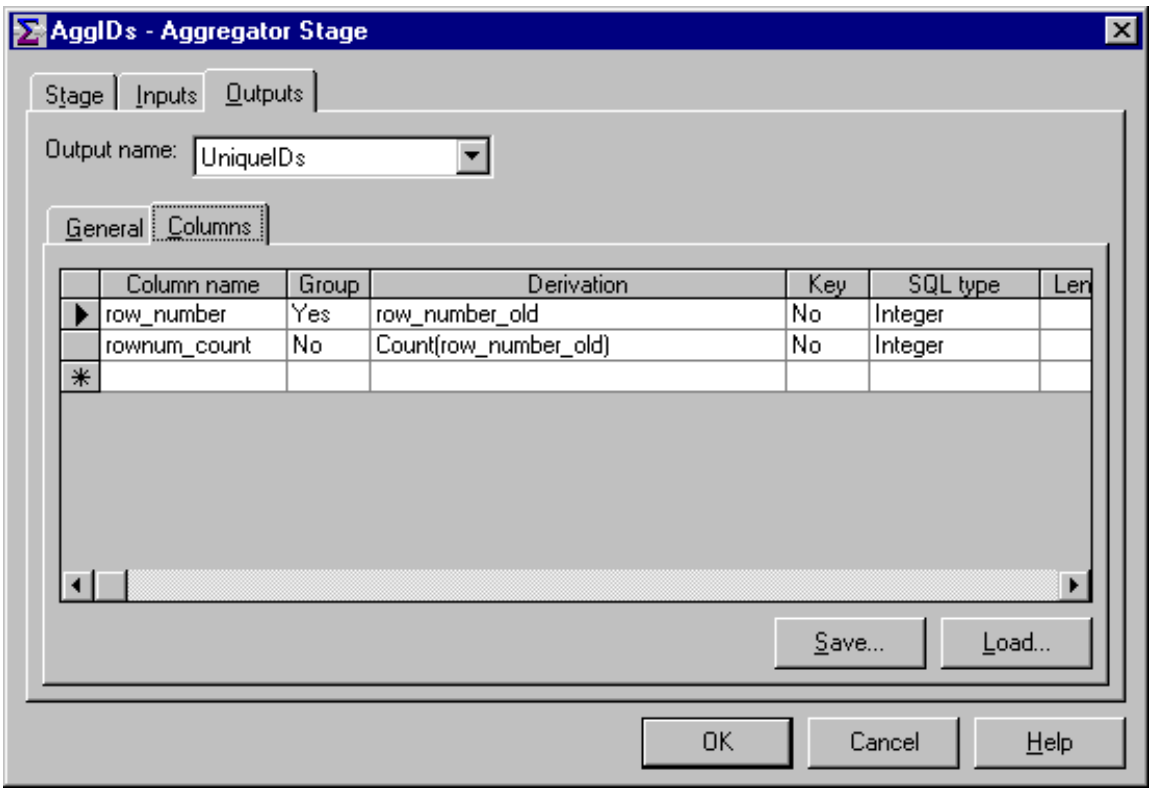
3a) Description of this stage.



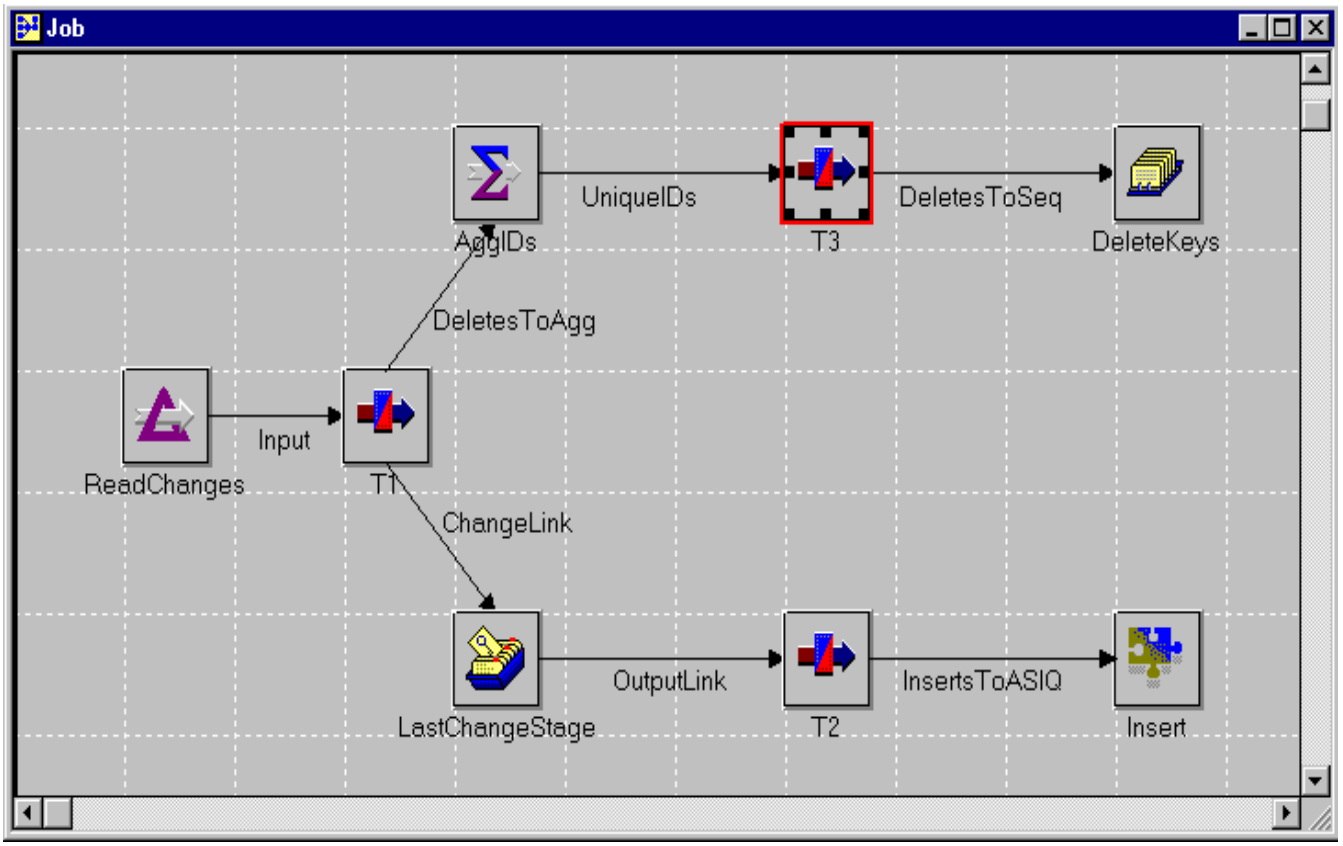
3b) Define the columns within the **Inputs** panel. **Note:** check how Display value is greater than Length, we ran into problems trying to use the same value so the only way we were able to Compile the Job was to increase the value for Display.



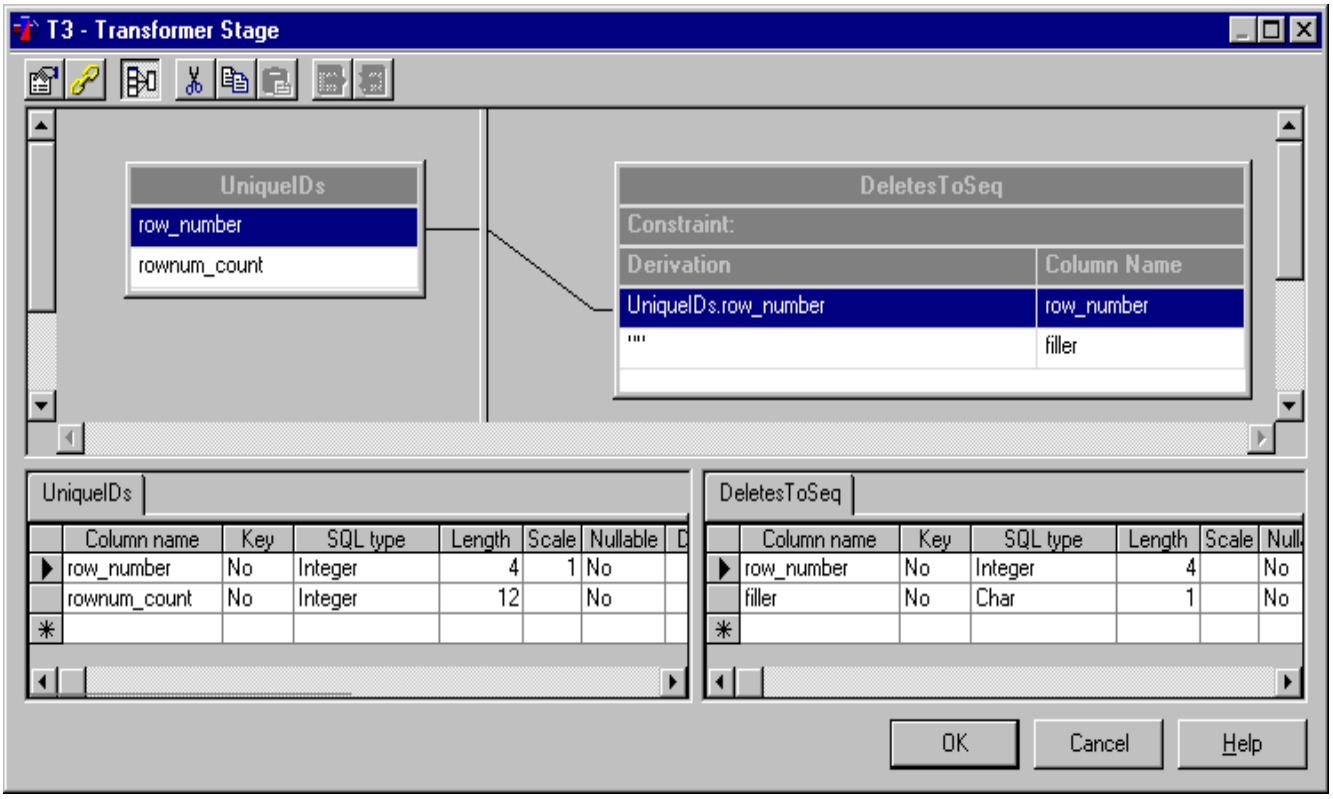
3c) As described before, the count operation is used to give the Aggregator a function to perform within the **Outputs** panel.



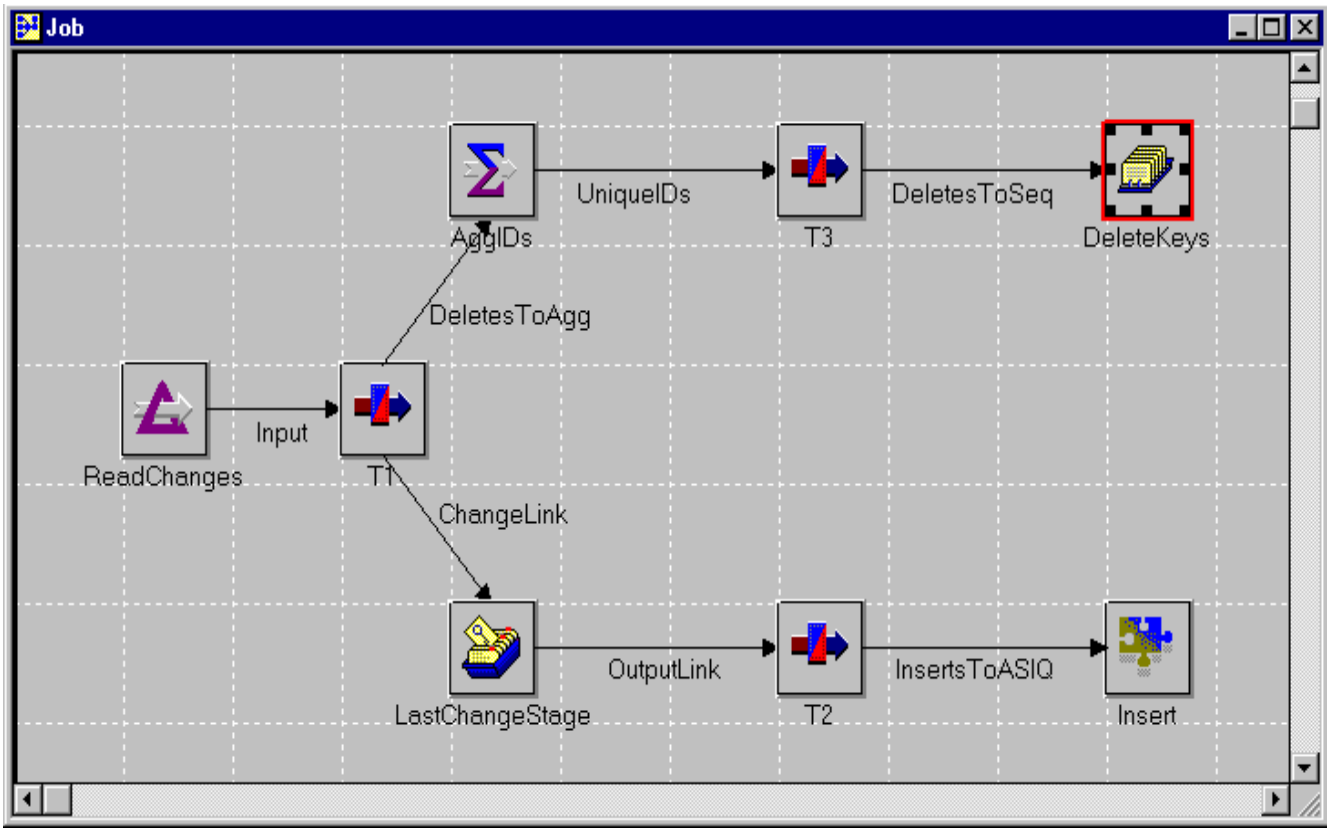
4.Transformation Stage (T3)



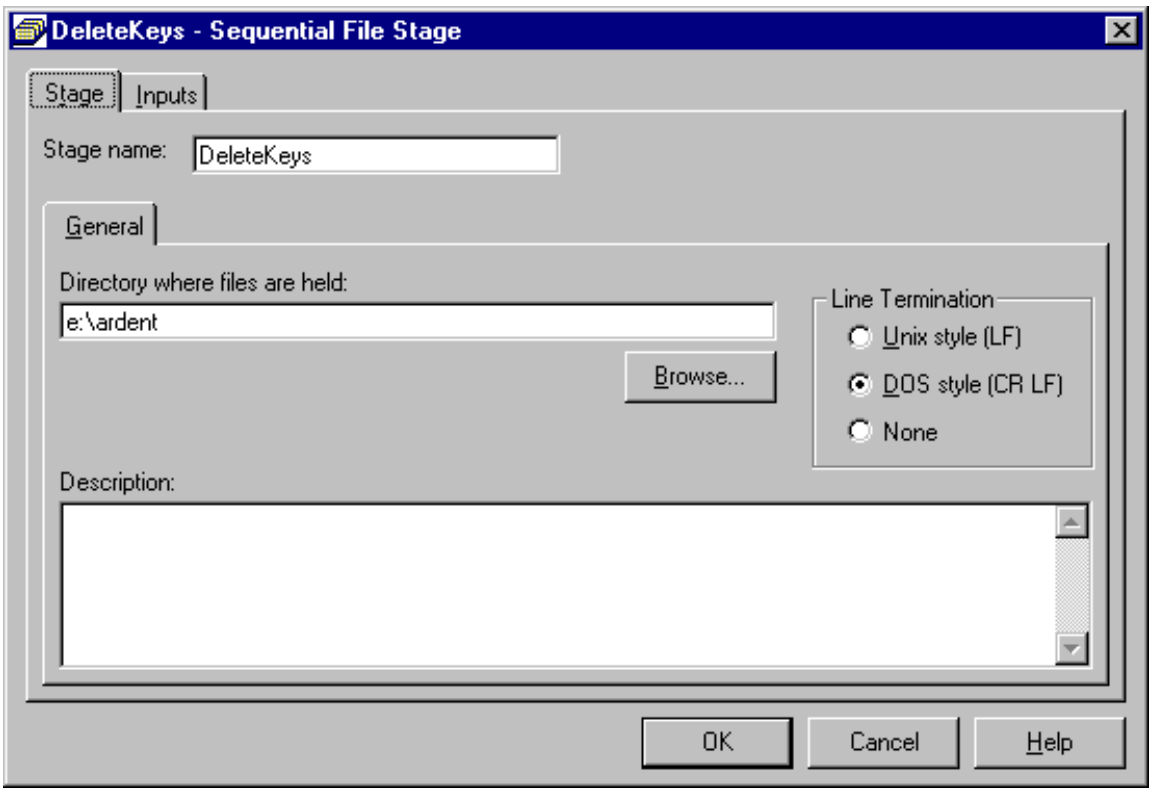
4a) Create the derivation between stages and make sure that for filler column you define a double ".



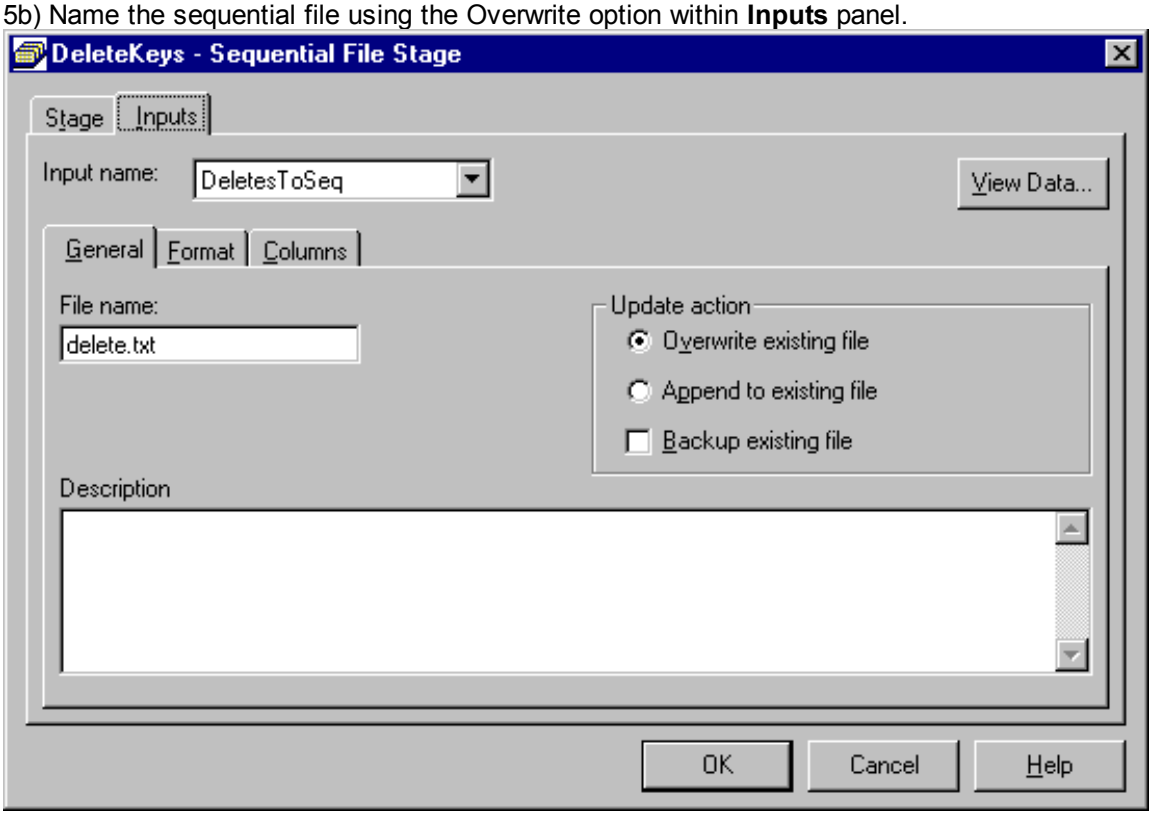
5. Sequential File Stage (DeleteKeys)



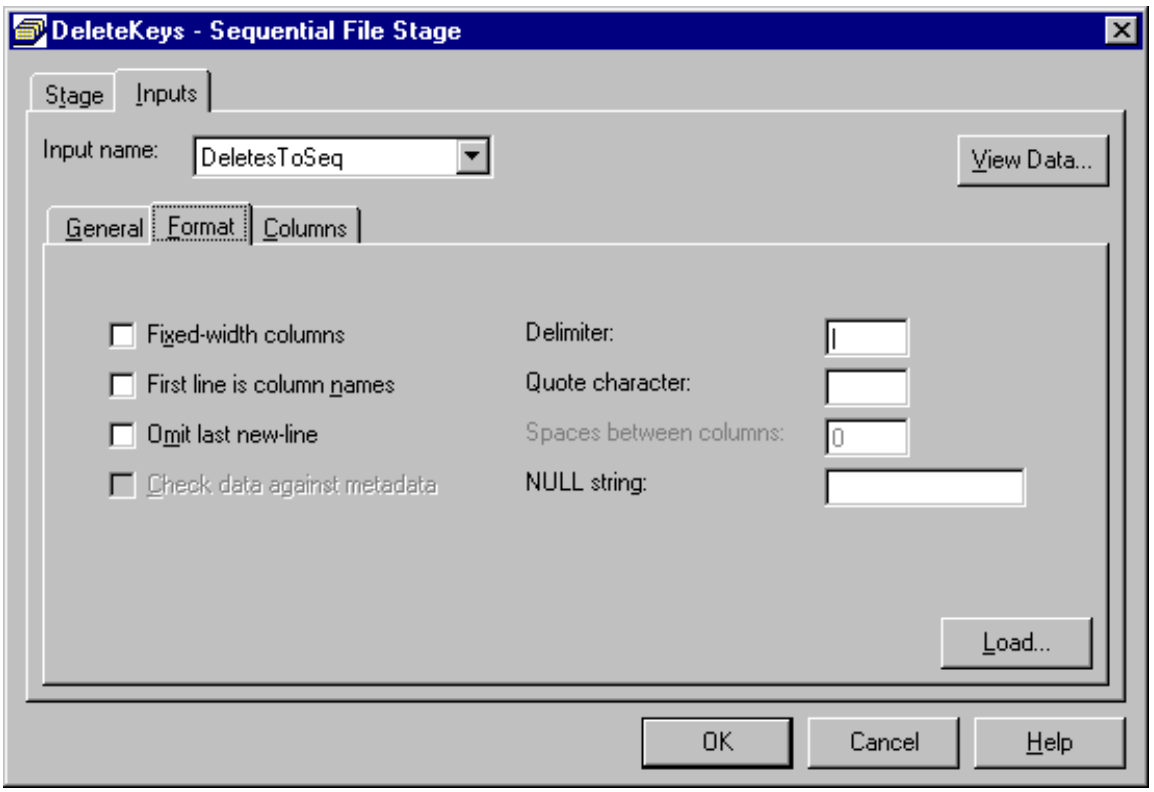
5a) For the Stage panel define the directory in which to store the sequential file.



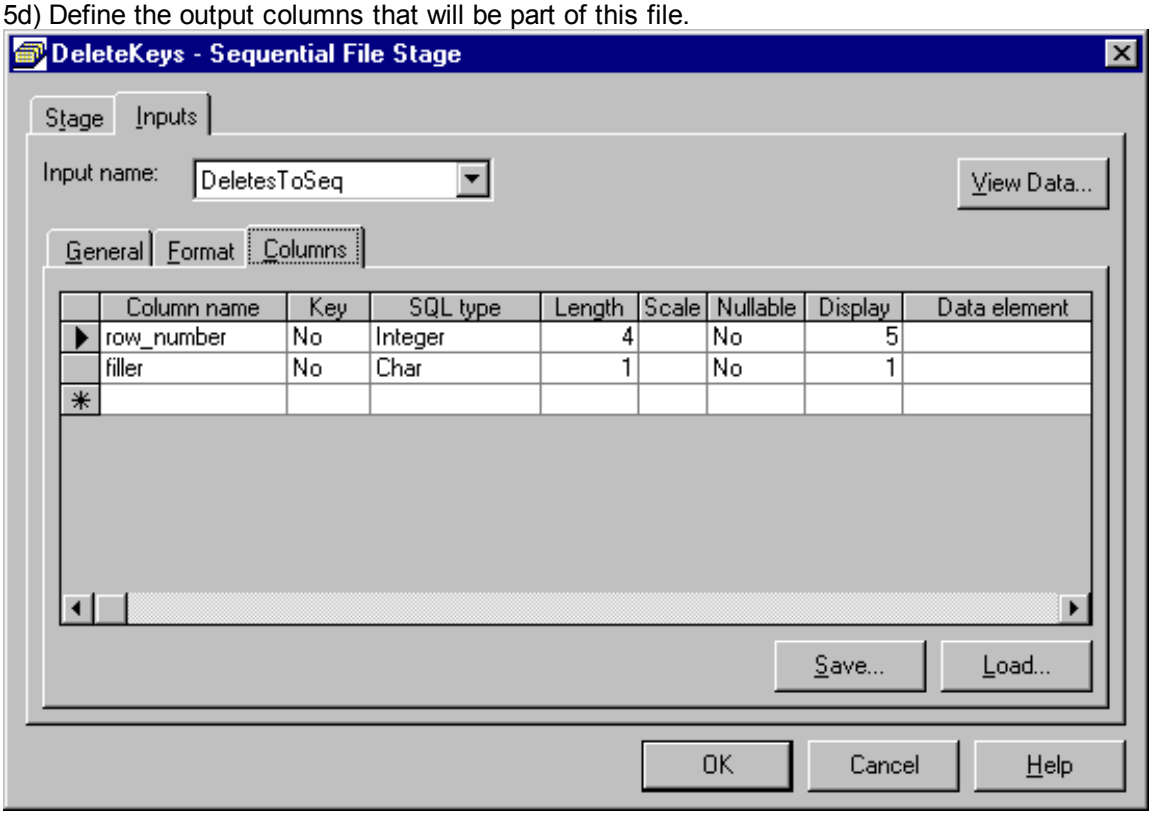
5b) Name the sequential file using the Overwrite option within **Inputs** panel.



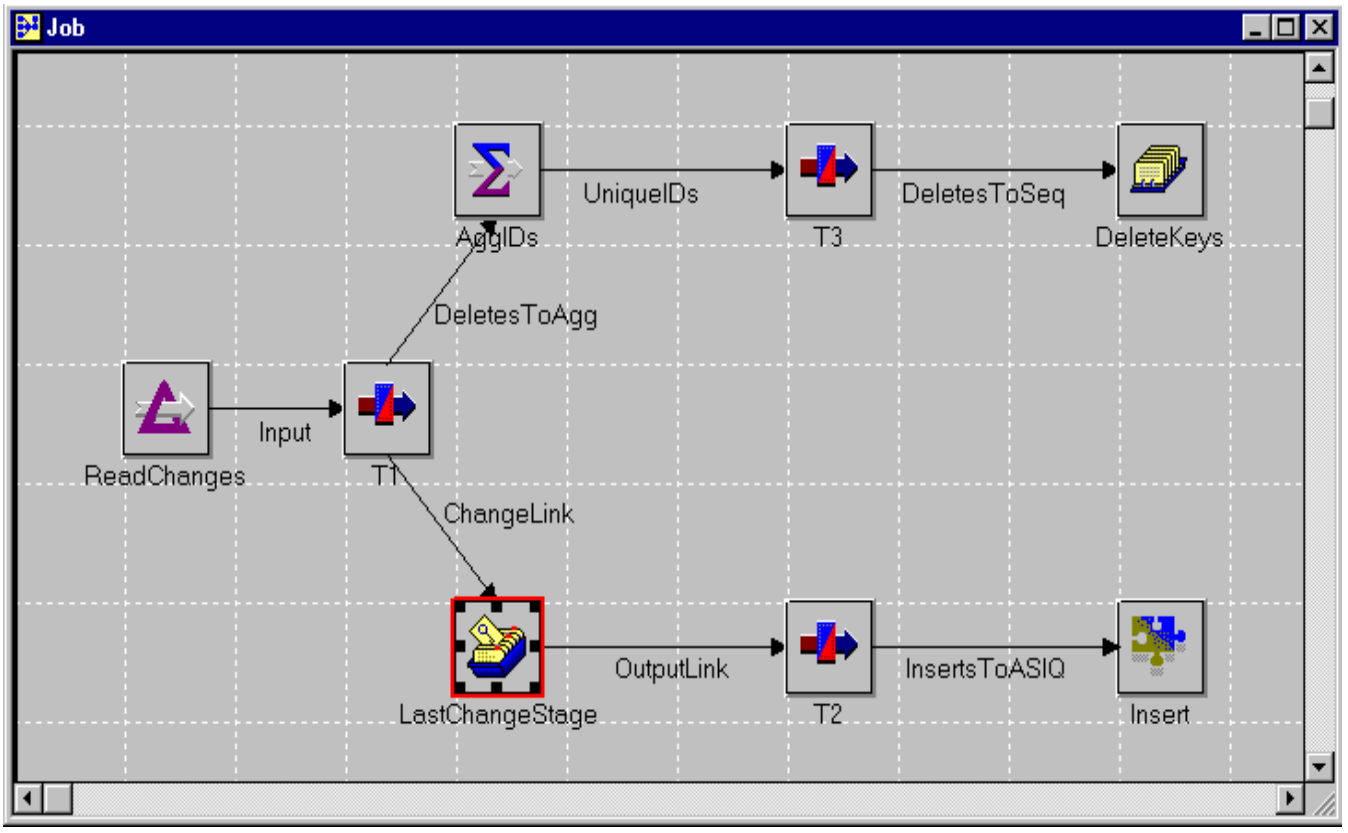
5c) Following are the format options used in the sequential file.



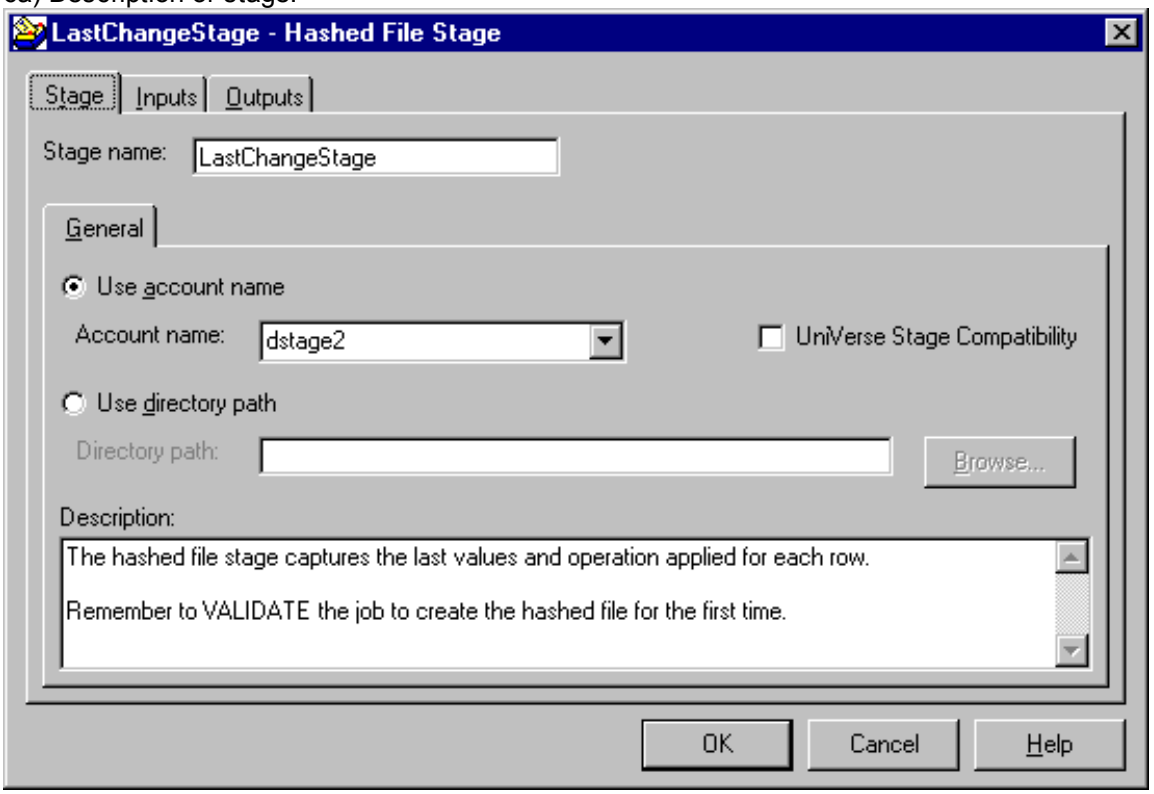
5d) Define the output columns that will be part of this file.



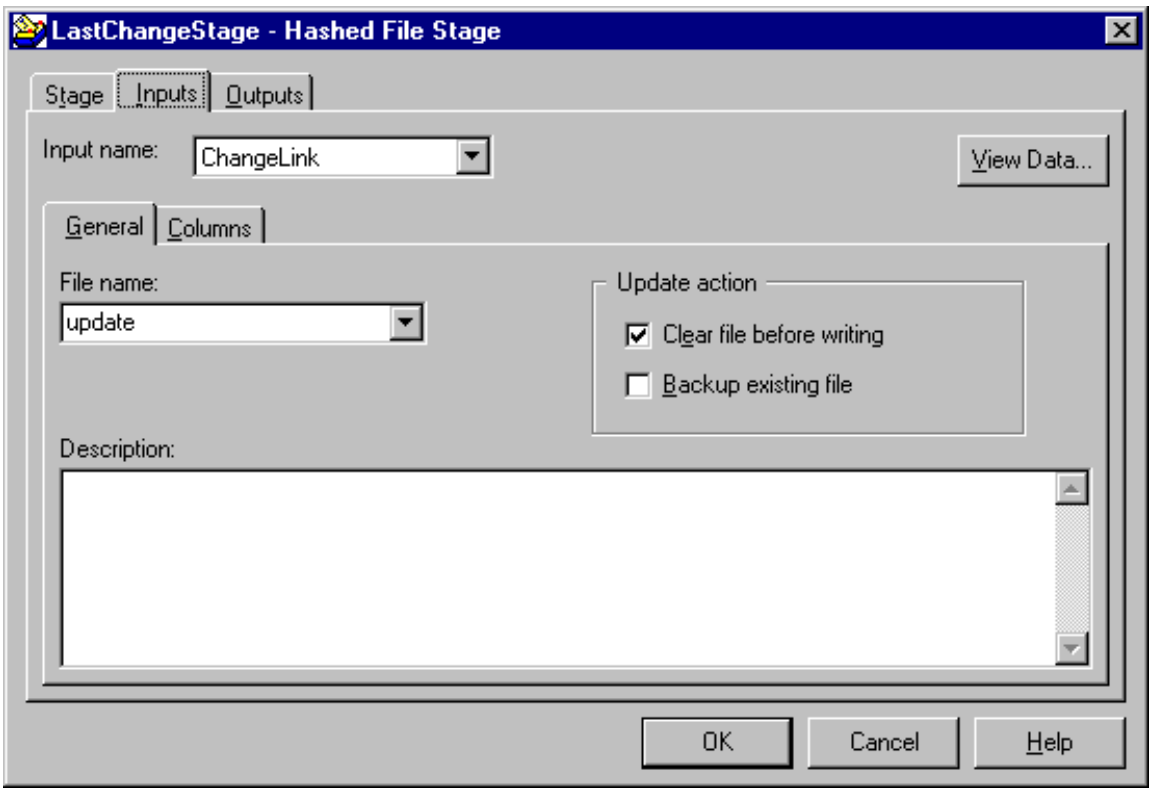
6. Hashed File Stage (LastChangeStage)



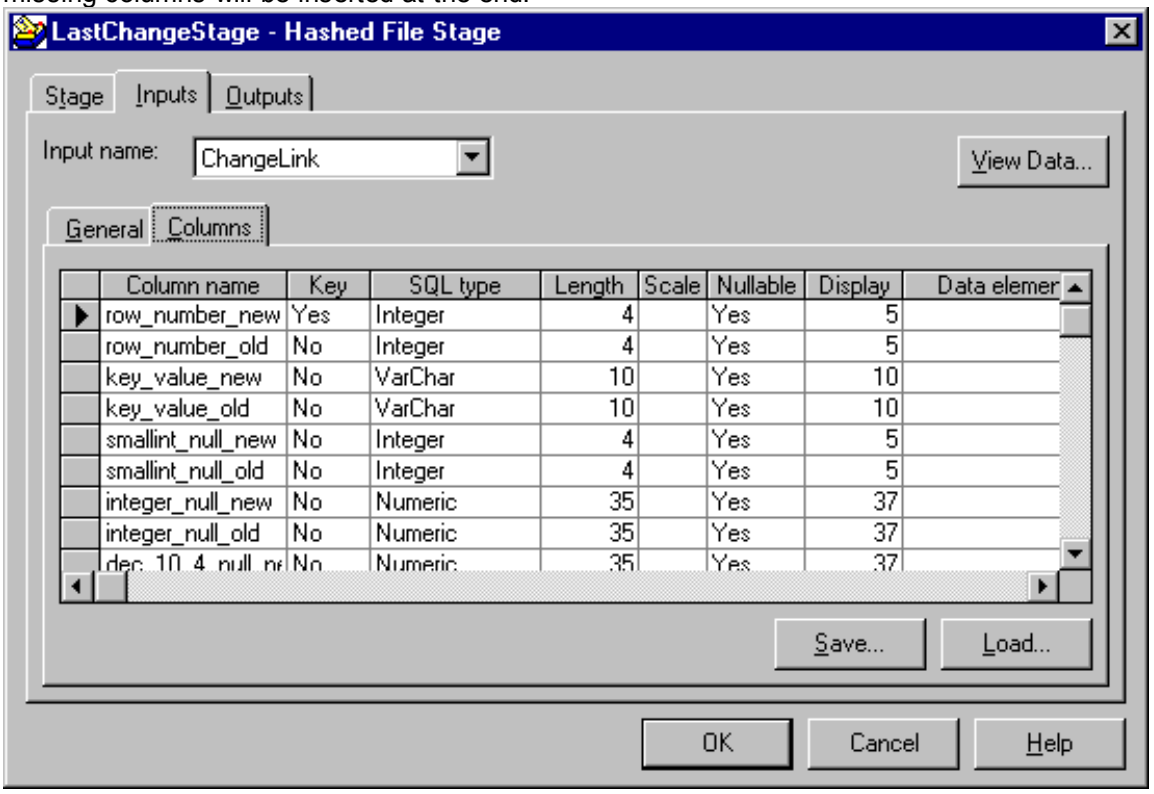
6a) Description of stage.



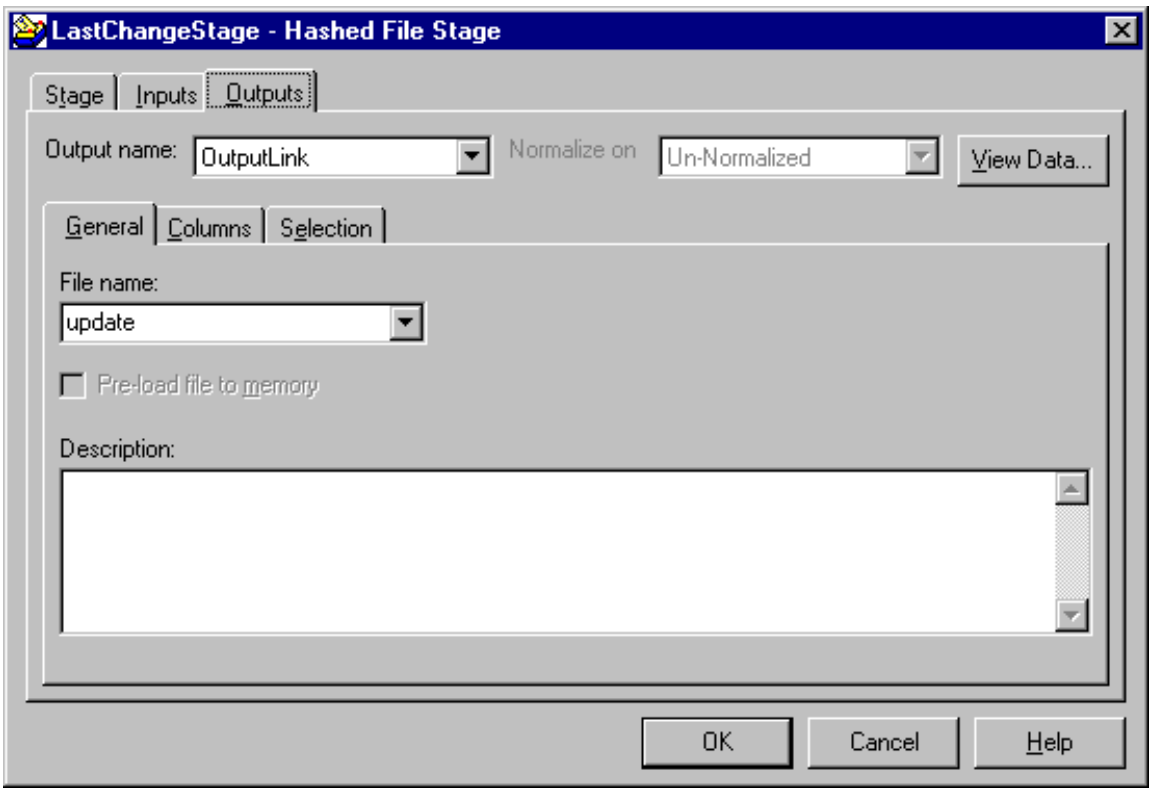
6b) Define the file name using "Clear file before writing" option.



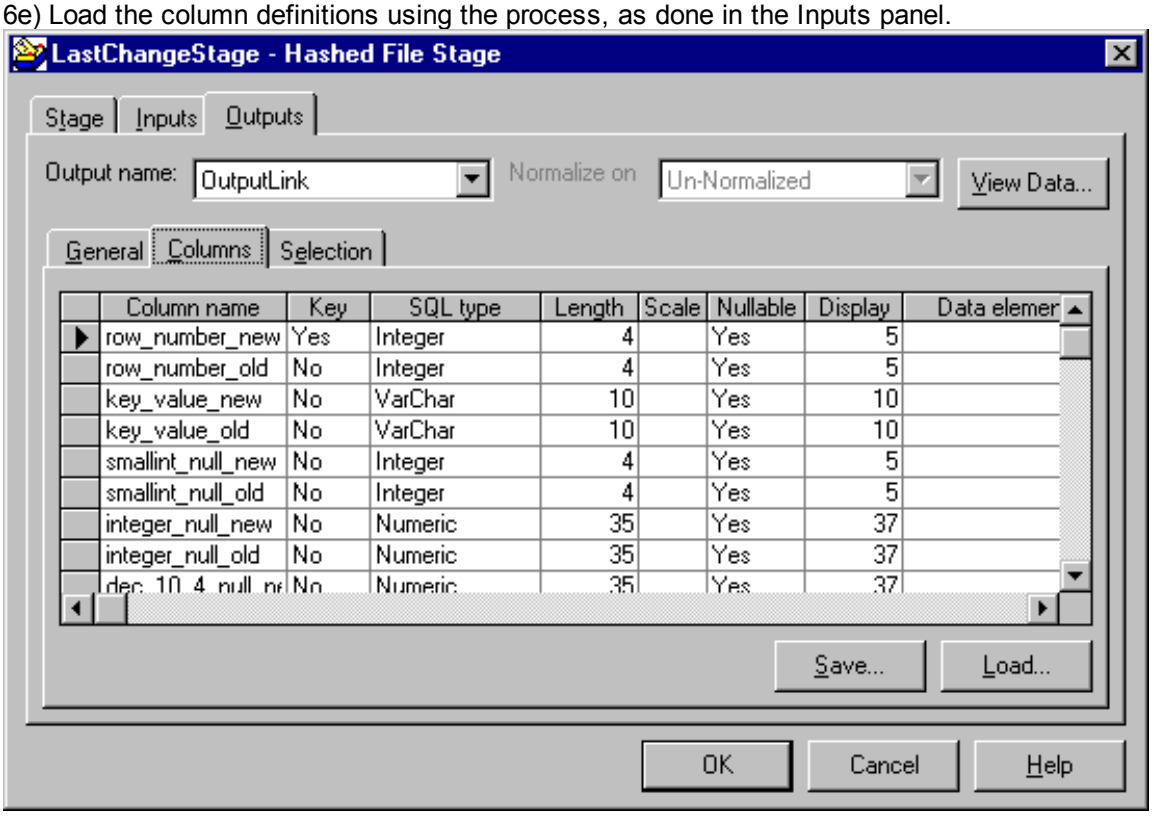
6c) Load the column definitions from Metadata. **Note:** check how the order in this file changes. The first column is "row_number_new" instead of "Timestamp" and the Key field is marked with "Yes". The reason is this is our key to select the last change for a specific row and a Timestamp field would not be useful for this purpose. A small tip for this operation is to use the same CDC metadata file that was saved at subscription definition within the CDC Configuration utility then just remove the first two columns (Timestamp and Operation) and reload the same metadata, since the rest of the columns have been already loaded. The two missing columns will be inserted at the end.



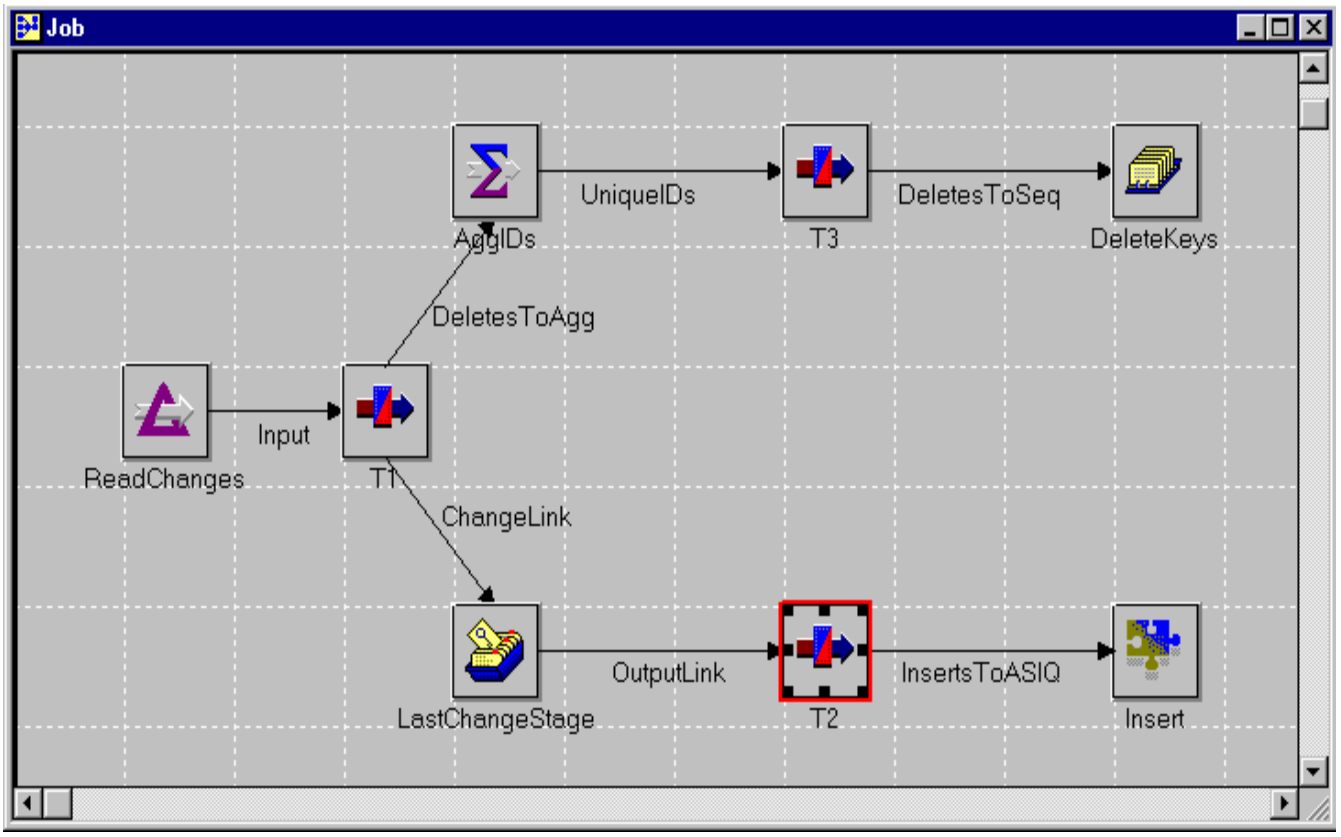
6d) Define the same file name on the **Outputs** panel.



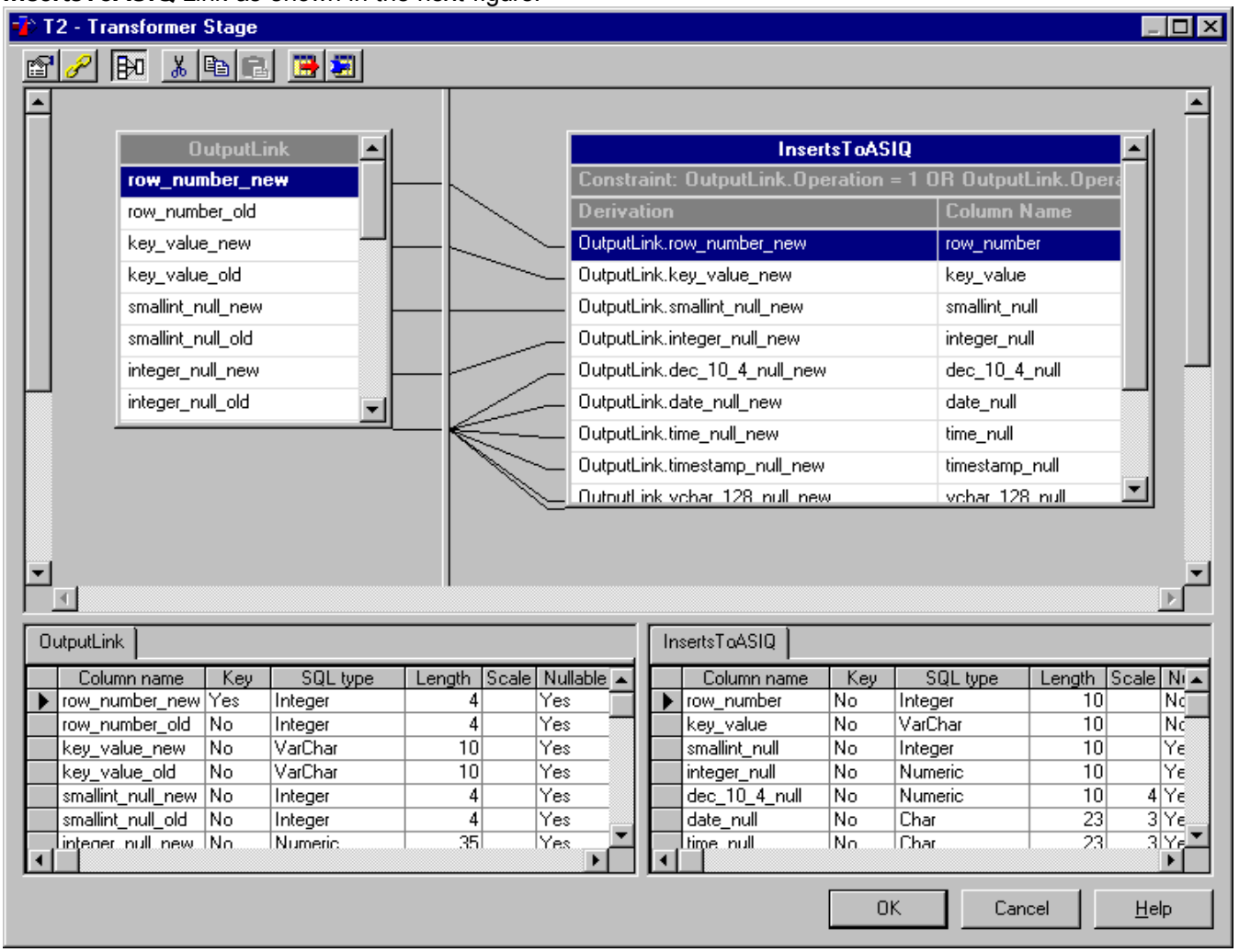
6e) Load the column definitions using the process, as done in the Inputs panel.



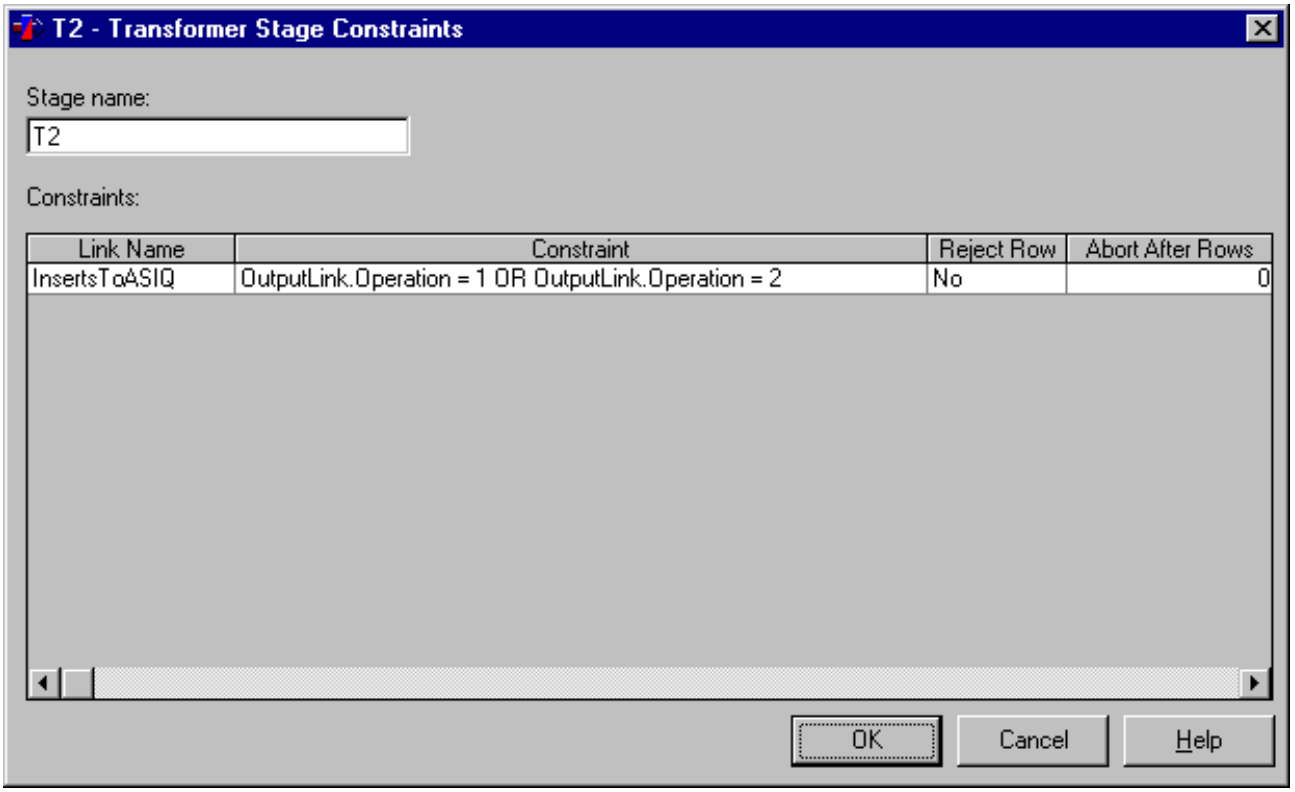
7. Transformation Stage (T2)



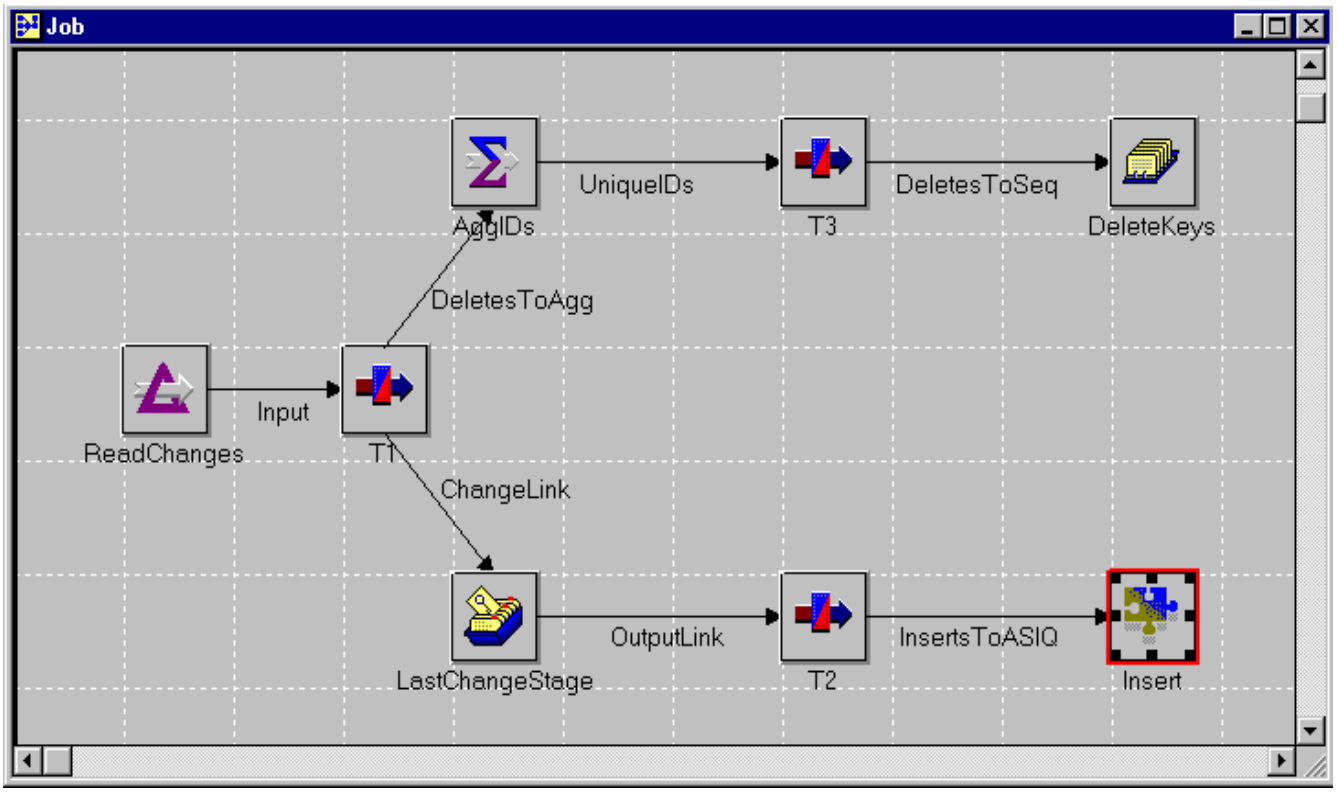
7a) Create the right derivation for each column. In this case, the new column values will be passed to the **InsertsToASIQ** Link as shown in the next figure.



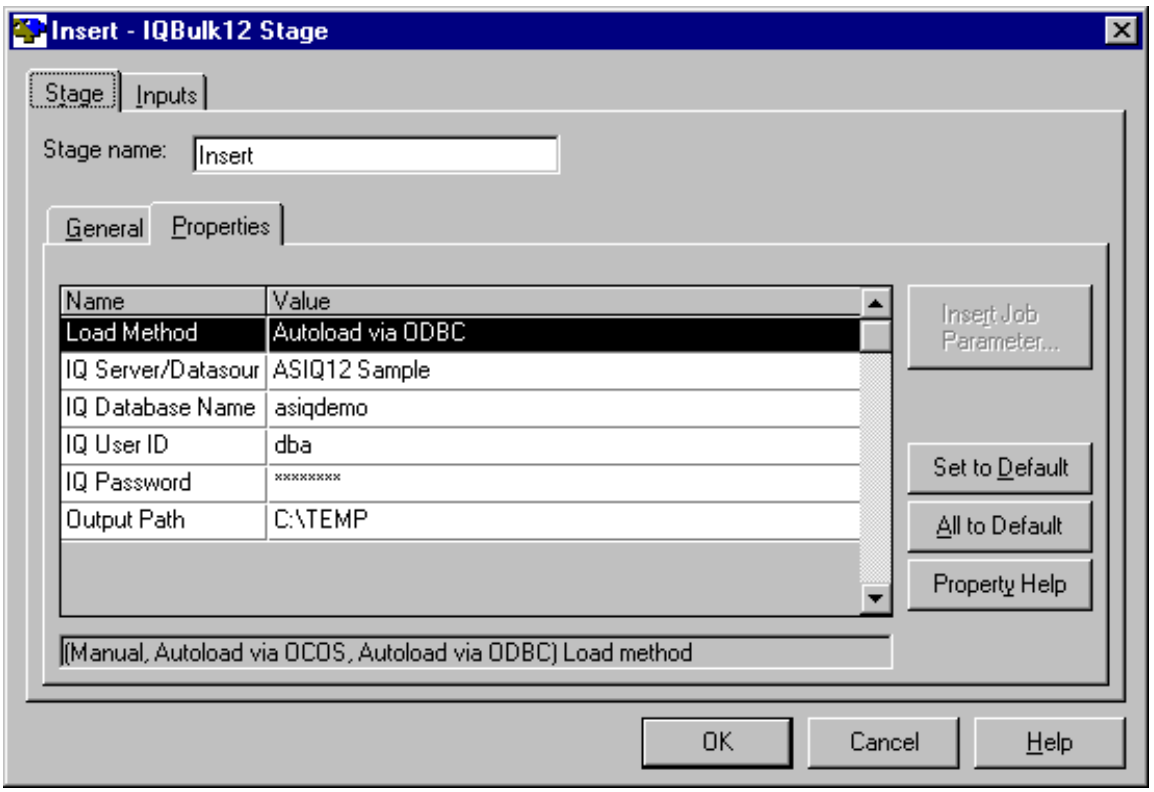
7b) The Constraint defined on this stage will allow only those operations marked as 1 (INSERT) and 2 (UPDATE) to be passed. In other words the DELETE statements are ignored in this stage.



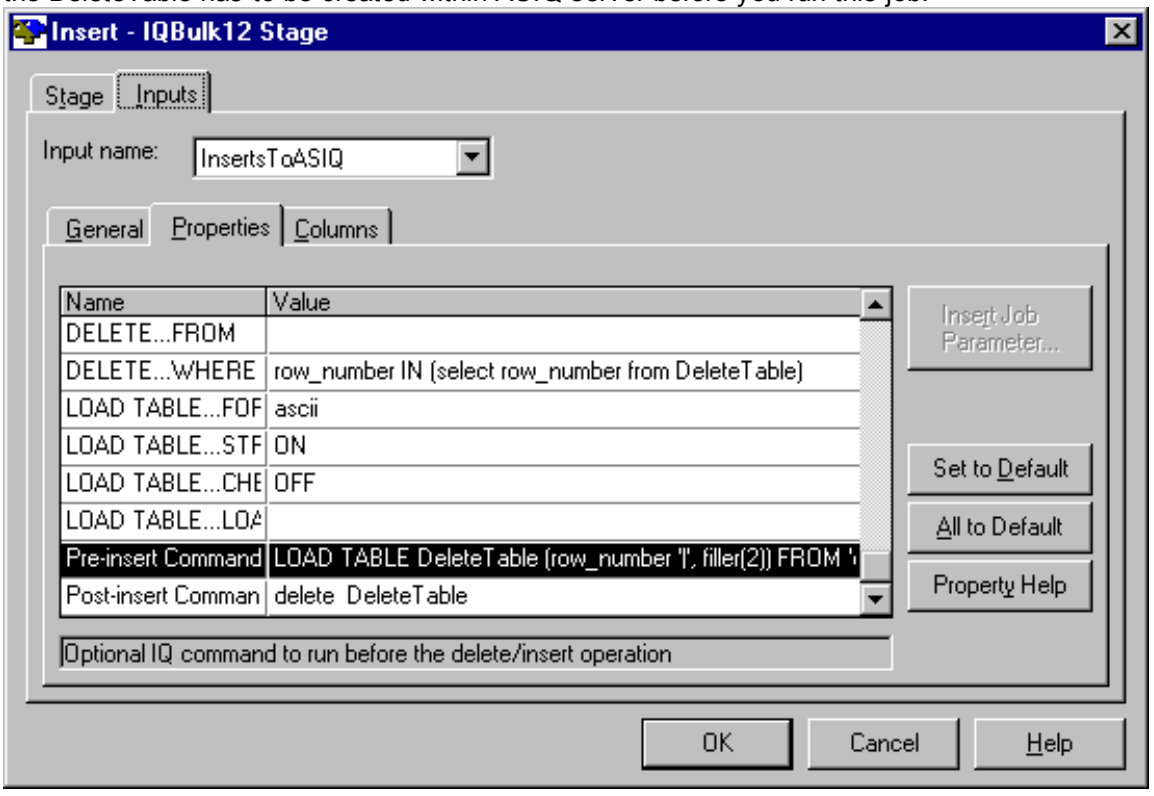
8. ASIQ Plug-in Stage (Insert)



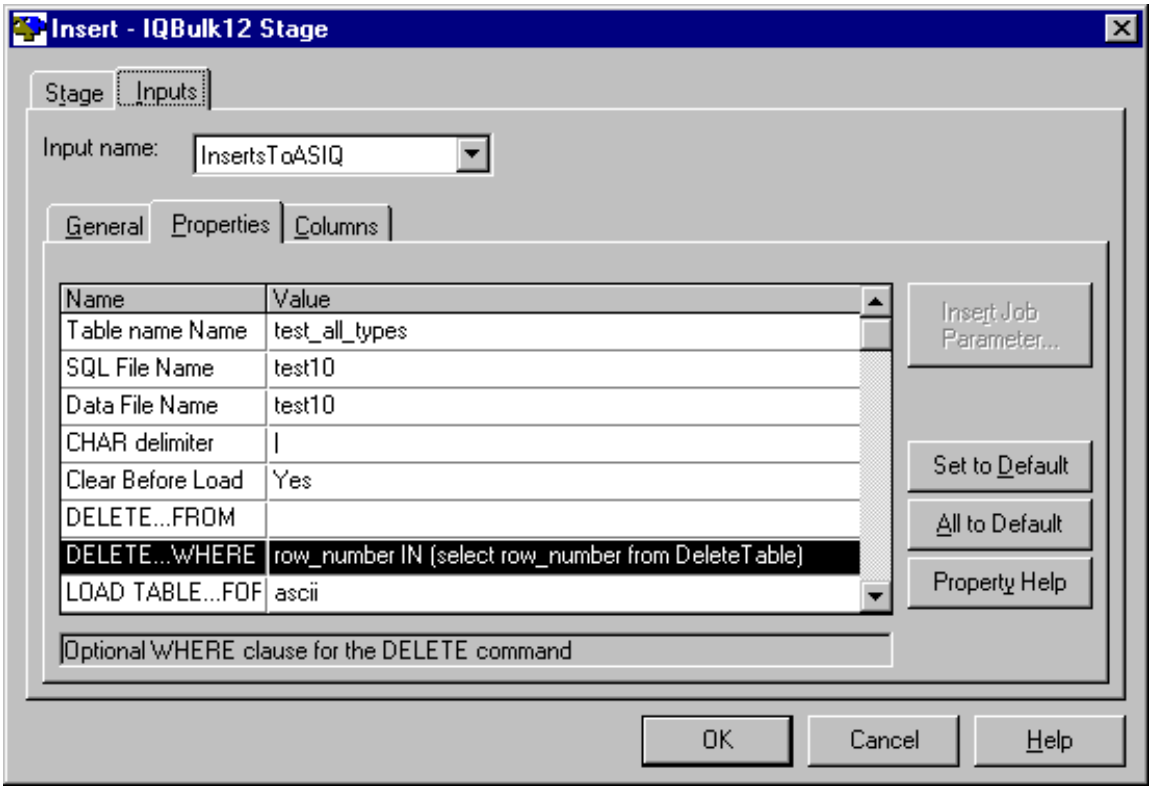
8a) In the **Stage Panel - Properties** we define the all the necessary information to make the connection for ASIQ using the ODBC driver.



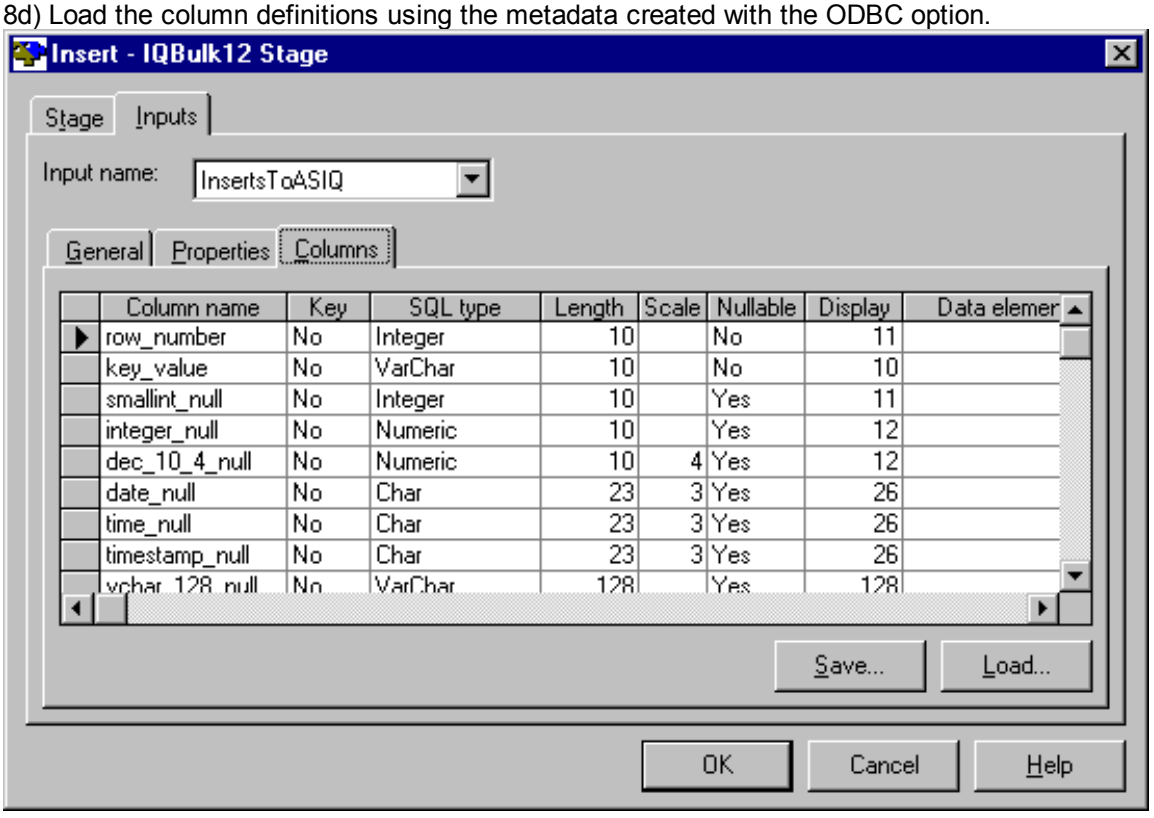
8b) This **Inputs -> Properties** panel handles the most important piece of this Job. Defined in the Pre-Insert Command option is the LOAD of the DeleteTable using the Delete Sequential file (previously defined). **Note:** the DeleteTable has to be created within ASIQ server before you run this job.



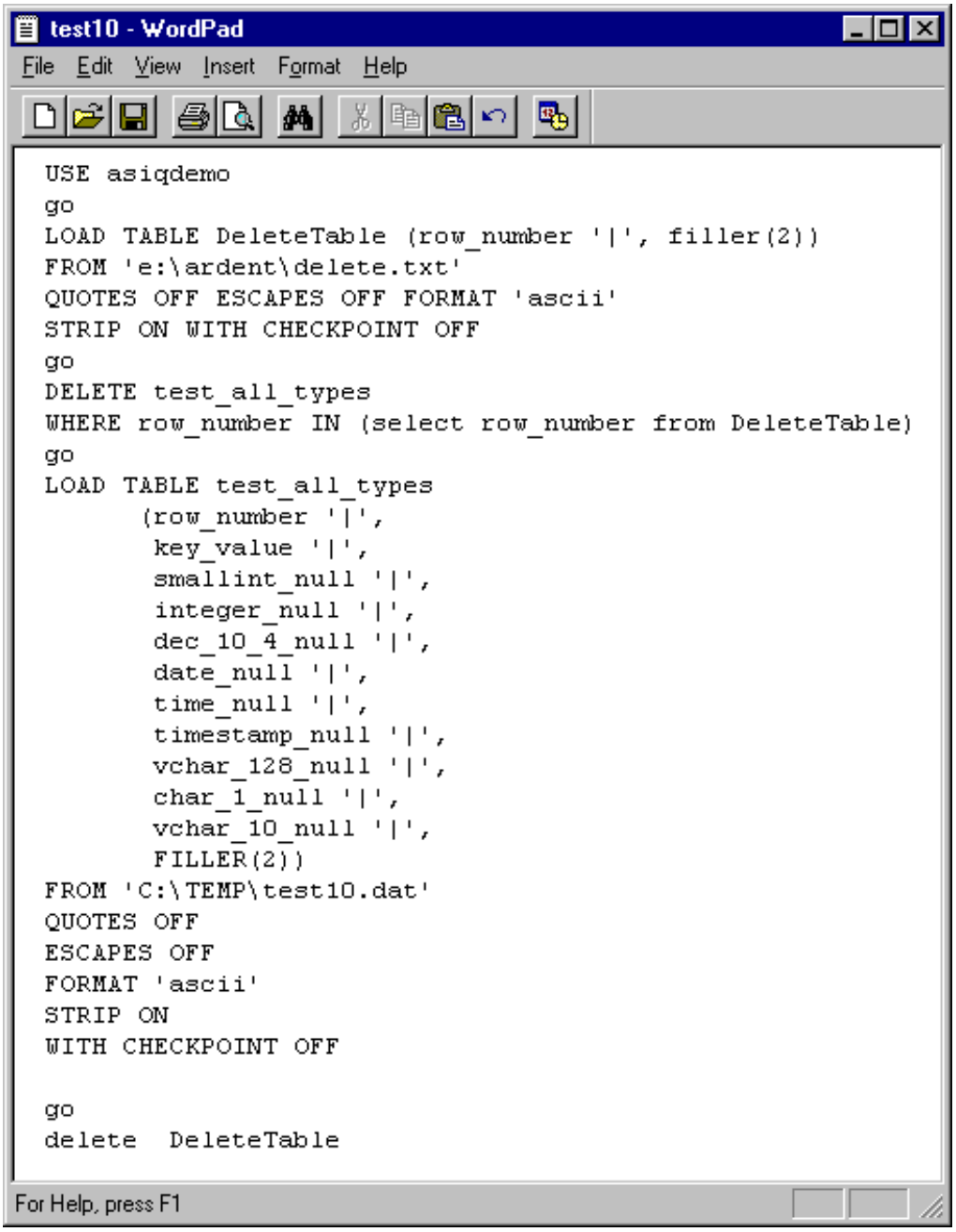
8c) After the LOAD of data for DeleteTable executes then the next DELETE sentence will execute (which deletes from the table the DELETE, UPDATE transactions) previous to the final LOAD that contains the INSERT, UPDATE transactions.



8d) Load the column definitions using the metadata created with the ODBC option.



8e) Save and compile the Job, and later run it using DataStage Director (to see more details on this you can check the Certification Report # 138 that describes in more detail the process of creating, compiling, verifying and running a Job with DataStage). In order to give you an idea, your SQL file (the one created for ASIQ Plug-in) should look pretty similar to the one used in our tests. Keep in mind that there could be other ways to create a Job that will use a different logic, but at the end the results should be identical.



This concludes the DataStage Job description to capture replicated events into ASIQ using Ardent's Change Data Capture Plug-in.