**Description of the Asset**:-

This asset is a part enhancement of IBM owned ETL tool Datastage for easing out a transformation using regular expression. As we know regex [regular expression] is big tool for cleansing , validation and transformation of unstructured data into a regular set but unfortunately Datastage as of now provide this facility inherently so we go for this enhancement using a parallel routine.

[NOTE: This Asset is not to be strictly made in public since this is an IBM internal property and already commercialized. So no public repository like GitHub upload has been done. ]

**Benefits of this Asset**:-

Using this asset you have the following benefit🡪

1. Portability :- You can use it in any project using datastage irrespective of much change to the code.
2. Flexibility :- You can use this routine and tweak it as per your requirement.
3. Usability :- Any kind of regex rule validation and regex match find can be done using routine.

Code:-

**checkRegularExpression**🡪



**checkRegularExpressionMatch**🡪



**Compilation Process**:-

Well the above code you need to place under /home/dsadm directory and give chmod 777 permission to the files. Then you need to run the below command to do the compilation.

g++ -c -O -fPIC -Wno-deprecated -m64 -mtune=generic -mcmodel=small checkRegularExpression.cpp

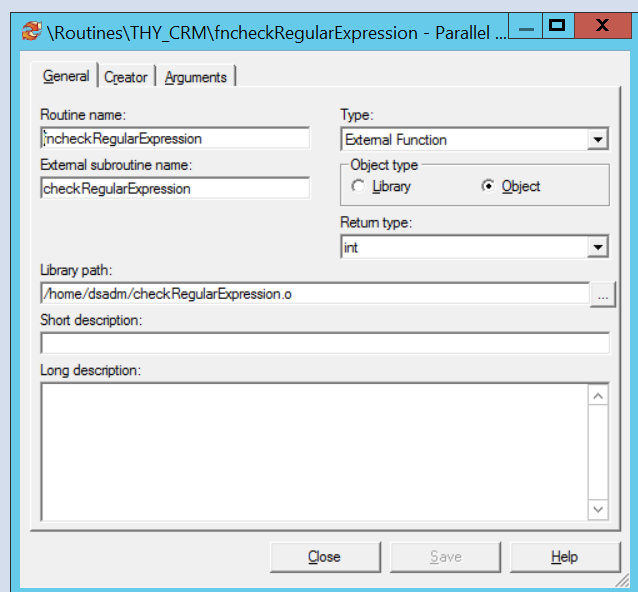
g++ -c -O -fPIC -Wno-deprecated -m64 -mtune=generic -mcmodel=small checkRegularExpressionMatch.cpp

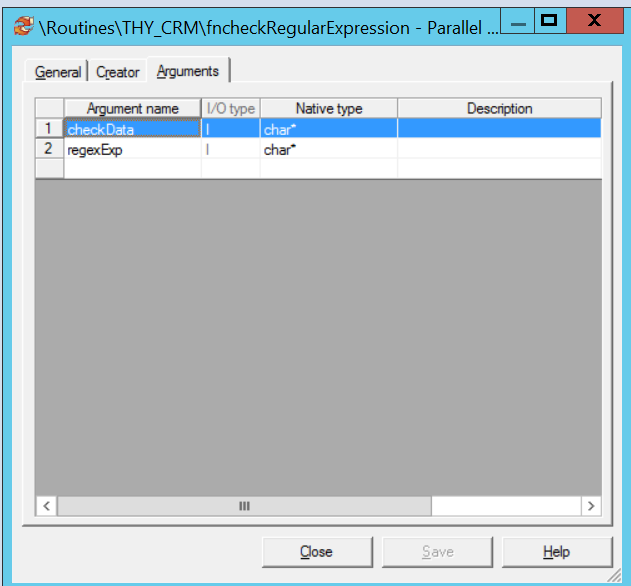
chmod 777 checkRegularExpression.o

chmod 777 checkRegularExpressionMatch.o

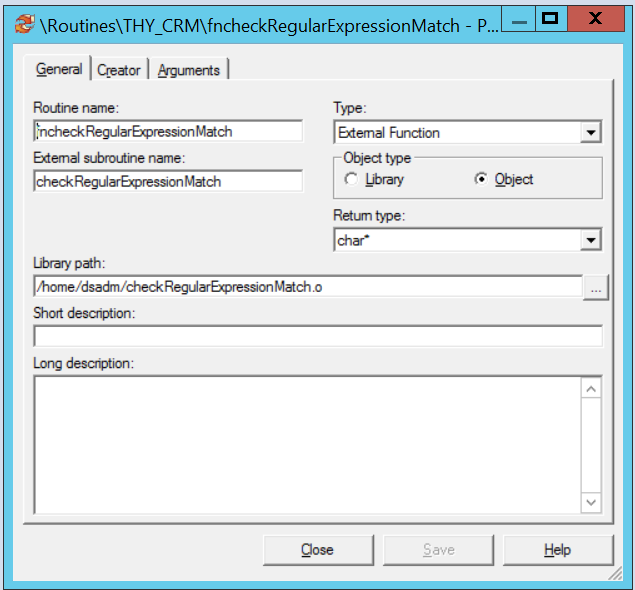
**Routine in Datastage**:-

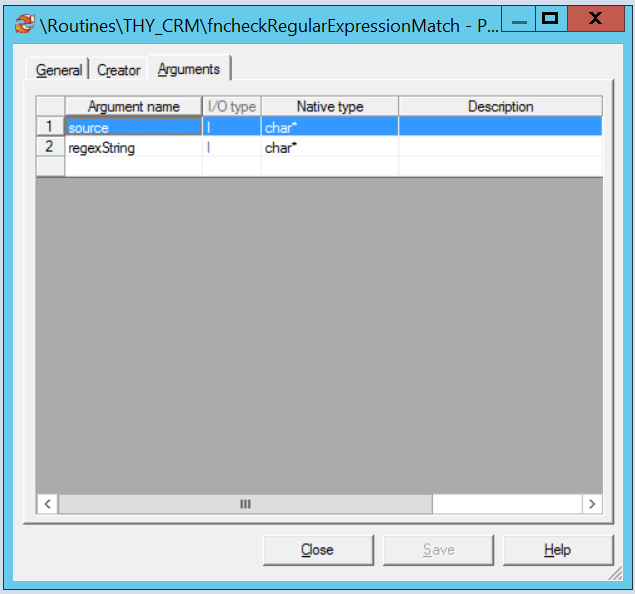
**fncheckRegularExpression**🡪



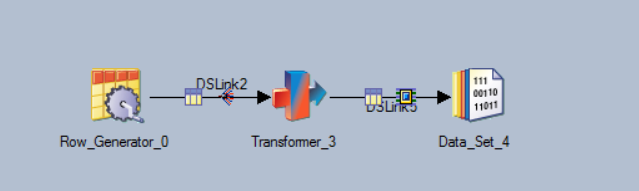


**fncheckRegularExpressionMatch** 🡪





**Usage Example in a Parallel Job:-**



Here in this job we are testing a regular expression for finding out the 1st occurrence of the numeric part from an unstructured data. “PH (+91) 033-2627 3111 ANUP 145/EN” this is our data and we supposed to extract the phone number from it. For the first step we are removing space,’+’,’)’,’(‘,’-‘ using datasatge convert function [**svPhone = Convert('+',"",Convert(')',"",Convert('(',"",Convert('-',"",Convert(' ',"",NullToEmpty(DSLink2.phone))))))**] to get a relatively cleansed data like PH9103326273111ANUP145/EN but from here it is still virtually impossible to extract the phone number which is 9103326273111. But here we can apply the regex rule ([0-9]+) which means the occurrence of a numeric string inside the alpha numeric string and using our above created routine we get the 1st occurrence of that.

[**NullToEmpty(fncheckRegularExpressionMatch(svPhone, "[0-9]+"))**]



**Conclusion**:-

Now you can use any expression instead of our above example [0-9]+ using this routine to take out your correct data from any unstructured unformatted text. This function can be used as an enhanced “Field()” function of Datastage as well if you want to find out a delimiter which is a regular expression.

So from overall utility perspective this asset is very useful and generic.