# Mobilize.Net SnowConvert for Spark

SnowConvert not only can execute migration, but reports on what it found when scanning the source codebase. In this detailed report, you will find a summary of the technical considerations and code analysis performed by SnowConvert when evaluating your source application. You will also find a high-level inventory and estimate of the automation provided by SnowConvert for the source codebase. If you have any feedback or any additional questions regarding the information below, please reach out to [support@mobilize.net](mailto:support@mobilize.net).

***Interpreting the Readiness Score***

Readiness is a measure of identifies references to the Spark API that can be converted divided by the total references found in this workload.

**Score** > 80% - Ready to migrate with minimal effort

60% > **Score** > 80% - Ready to migrate with some manual effort required

**Score** < 60% - Additional information will be required

***Understanding the Conversion Score***

The Conversion Score takes all usages of the Spark API that can be converted by SnowConvert divided by all references found in this workload.

**S**PARK SCALA API SUMMARY

|  |  |
| --- | --- |
| Identified Usages: | 264 |
| **READINESS SCORE:** | 82.58% ± 1.34% |
| Usages Ready for Conversion: | 218 |
| **CONVERSION SCORE:** | 82.58% ± 1.34% |
| Usages Converted: | 218 |
|  |  |
| ± Margin of error. If there were files with parsing errors, this would affect the accuracy of the readiness score and conversion score. For that reason, the tool introduces a margin of error that must be considered. | |

## EXECUTION SUMMARY

### Customer

OwnerName not provided

OwnerCompany not provided

OwnerEmail not provided

### **Session** **ID**

589b8506-ce73-40ee-9bc0-d938649a9163

### Timestamp

2023/03/24 05:41:40

### Version Information

SnowConvert for Spark: V1.7.0  
Snowpark Library for Scala: V1.01.039

## FILE SUMMARY

SnowConvert for Spark scans all files that are present in your source codebase. This includes files that are not just Scala, but any other files. You can find a summary of the files, lines of code, and count of files with Scala usages present in the codebase.

### File Type Summary

|  |  |  |  |
| --- | --- | --- | --- |
| **Technology** | **Lines of Code** | **File Count** | **Percentage of Total Files** |
| Scala | 33152 | 186 | 72.37% |
| Other\* | N/A | 71 | 27.63% |
| **Total** | **33152** | **257** |  |

*\** *Other refers to technologies not recognized as code*

### Scala File Content

|  |  |  |
| --- | --- | --- |
| **Technology** | **File Count** | **Percentage of all Scala Files** |
| Spark usages | 99 | 53.23% |
| Embedded SQL | 0 | 0.00% |

If there are no files with Spark usages, then the tool will not generate the Spark Reference Summary spreadsheet in the Reports folder in the output directory.

### Scala File Sizing

File sizing can help give you an idea of how complex a migration could be. The longer the file, the more likely there will be an issue in conversion.

|  |  |  |  |
| --- | --- | --- | --- |
| **Size** | **Abbreviation** | **File Count** | **Percentage of all Python Files** |
| Extra Small (less than 50 LOC) | XS | 175 | 68.09% |
| Small (less than 200 LOC) | S | 61 | 23.74% |
| Medium (less than 500 LOC) | M | 17 | 6.61% |
| Large (greater than 500 LOC) | L | 4 | 1.56% |

## SPARK USAGE SUMMARY

Spark usages are references in the source codebase to the Spark API. These are divided into several categories based on the Spark library. Supported usages are references to the Spark API that have a known conversion to the Snowpark API. Unsupported usages do not yet have an equivalency in the Snowpark API. You can find a summary of these with the readiness score by category below.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Usages Count** | **File Count** | **Percentage of all Scala Files** |
| Supported usages | 218 | 99 | 53.23% |
| Unsupported usages | 46 | 13 | 6.99% |
| **Total** | **264** | **99** | **-** |

### Spark Usage Summary

|  |  |  |  |
| --- | --- | --- | --- |
| **Spark API Usages** | **Supported Usages** | **Unsupported Usages** | **Readiness Score** |
| Dataframe | 72 | - | 100.00% |
| Column | 14 | - | 100.00% |
| Functions | 34 | - | 100.00% |
| RDD | - | - | - |
| SparkSession | 1 | - | 100.00% |
| Others | 97 | 46 | 67.83% |
| **Total** | **218** | **46** | **82.58%** |

|  |
| --- |
| Overall Readiness Score: **82.58% (218/264)** |

This readiness score is an estimate of the total quantity of reference to the Spark API divided by the total number of supported references from Spark to the Snowpark API.

### Spark Usages by Support Category

SnowConvert for Spark divided supported Spark elements into several categories based on the kind of mapping that is present from Spark to Snowpark. You can find more information about each of these categories in the appendixes at the end of this document.

|  |  |  |
| --- | --- | --- |
| **Status Category** | **Count** | **Percentage of Total References** |
| Direct | 206 | 78.03% |
| Rename | 12 | 4.55% |
| Helper | 0 | 0.00% |
| Transformation | 0 | 0.00% |
| WorkAround | 0 | 0.00% |
| NotSupported | 0 | 0.00% |
| NotDefined | 46 | 17.42% |
| **Total** | **264** | **-** |

*\* Check* ***appendix A*** *for more information.*

## SCALA IMPORT CALLS

All references to an external library in this codebase’s Scala code can be found here. These are not limited to calls to the Spark library, but any library references by the file.

|  |
| --- |
| Total Count of Scala Import Calls: **1453** |

### Scala Import Call Summary

|  |  |  |
| --- | --- | --- |
| **Packages** | **Count** | **Percent of All Import Calls** |
| org.apache.spark | 219 | 15.07% |
| com.amazonaws | 0 | 0.00% |
| java.io | 49 | 3.37% |
| Other | 1185 | 81.56% |
| Total | 1453 | 100% |

## SNOWCONVERT ISSUE SUMMARY

As a part of the conversion process, SnowConvert generates issues to highlight any errors or warnings that may be helpful when completing the migration process. Each of these errors and warnings have an error code associated with them. Each error code has a link to the page in our documentation that describes the error and gives recommendations on what you can do to resolve this error.

### Issue Categorization

Issues in Mobilize.Net SnowConvert are broken down into three categories:

|  |  |  |
| --- | --- | --- |
| **Warnings** | **Conversion Issues** | **Parsing Issues** |
| Code that is recognized and converted, but needs review. | Code that is recognized, but not converted. | Code that is not recognized and not converted. |

### Issue Summary

Below is a count of the issues present. You can find more information on each unique issue by clicking the link to each error code listed below.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Warnings** | **Conversion Issues** | **Parsing Issues** |
| # of issues | 0 | 46 | 5 |
| # of unique issues | 0 | 1 | 1 |

*\* Check* ***appendix B*** *for more information.*

### Issue List

|  |  |  |
| --- | --- | --- |
| **Code** | **Description** | **Instances** |
| SPRKSCL1142 | Spark element is not supported | 46 |
| SPRKSCL1001 | This code section has parsing errors, so it was commented out | 5 |

# APPENDIXES

## APPENDIX A: SPARK REFERENCE CATEGORIES

SnowConvert for Spark divided supported Spark elements into several categories based on the kind of mapping that is present from Spark to Snowpark. Below is a summary of each of the categories that SnowConvert outputs to describe the translation of each Spark reference, along with a description, example, and whether the tool can automatically convert the reference (Tool Supported) and if it’s possible the Snowpark

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Status** | **Description** | **Snowpark Supported** | **Tool Supported** | **Spark Example** | **Snowpark Example** |
| **Direct** | Direct translation. The same function exists in Spark and Snowpark with no change needed. | **TRUE** | **TRUE** | col("col1") | col("col1") |
| **Rename** | The function from Spark exists in Snowpark, but there is a rename that is needed. | **TRUE** | **TRUE** | val x = col("col1") collect\_list(x) | val x = col("col1") array\_agg(x) |
| **Helper** | The function from Spark have small difference in Snowpark than can be addressed by creating a function with an equivalent signature at a helper file created by Mobilize that will resolve the difference. Examples of this are "fixed" additional parameters, change order of parameters, etc. | **TRUE** | **TRUE** | asc(colname) | creating a helper function named asc with a identical signature as the spark function, like:  def asc(colname: String) = col(colname).asc |
| **Transformation** | The function is completely recreated to a functionally equivalent function in Snowpark, but doesn't resemble the original function. This can include calling several functions, or adding multiple lines of code. | **TRUE** | **TRUE** | var x : Column = col("col1") struct(lit(x).as("ROW")) | var x : Column = col("col1") object\_construct(lit("ROW"), lit(x)) |
| **WorkAround** | This category is employed when the tool cannot convert the Spark element but there’s a known manual workaround to fix the conversion (the workaround is published in the tool documentation) | **TRUE** | FALSE | val x = col("col1")x.otherwise("test") | val x = col("col1")  x.otherwise(lit("test")) |
| **NotSupported** | This category is employed when the tool cannot convert the Spark element because there's no applicable equivalent in Snowflake. | FALSE | FALSE | val sample = sampleDf.join(broadcast(SAMPLE\_TEST),   Seq("NAME"), "left") | val sample = sampleDf.join(/\*EWI: SPRKSCL1101 => Broadcast is not supported\*/ broadcast(SAMPLE\_TEST), Seq("NAME"), "left"). |
| **NotDefined** | This category is employed when the tool detects the usage of a Spark element as such but cannot be converted because it is not in the tool's conversion database. | FALSE | FALSE | NA | NA |