**Thomas McGee DIS Conversion Tool**

**Requirement Document**

**(DRAFT)**

|  |  |  |
| --- | --- | --- |
| **Template Revisions** | **Date** | **Change** |
| V1.0 | 01/027/2012 | Initial Version |
|  |  |  |
|  |  |  |

© 2012 CSC. Falls Church, Virginia. All rights reserved. No part of this publication may be reproduced by any means without written permission from CSC. Printed in U.S.A. All questions regarding this documentation should be routed through customer assistance, Blythewood, SC, 800-420-6007 or email [pcssupport@csc.com](mailto:pcssupport@csc.com).

#### Table of Contents

[1. Requirement Overview 4](#_Toc229538626)

[1.1. Assumptions and Constraints 4](#_Toc229538627)

[1.2. Current Scenario 4](#_Toc229538628)

[1.3. Proposed Scenario 4](#_Toc229538629)

[1.4. Functions to be addressed 4](#_Toc229538630)

[2. Requirement 4](#_Toc229538631)

[2.1. Functional Considerations 5](#_Toc229538632)

[2.2. Design Considerations 5](#_Toc229538633)

[2.3. Interface Considerations 5](#_Toc229538634)

[2.3.1. System Interfaces 5](#_Toc229538635)

[2.3.2. Hardware Interfaces 5](#_Toc229538636)

[2.3.3. Software Interfaces 6](#_Toc229538637)

[2.3.4. User Interfaces 6](#_Toc229538638)

[2.4. Performance Considerations 6](#_Toc229538639)

[3. Dependencies between Requirements 6](#_Toc229538640)

[4. Approach 7](#_Toc229538641)

[5. Signoff 8](#_Toc229538642)

[6. References 9](#_Toc229538643)

[7. Requirement Change Log 10](#_Toc229538644)

# 

# 1. Requirement Overview

In order to facilitate Thomas McGee’s (the client) use of the Data Analytics (DA) Data Import System (DIS) Employee import module to import employee information received from one of their vendors, a tool must be created to convert the data received from this vendor into a format acceptable to DIS. This tool will only be used to convert the client’s file from its original format to one that can be accepted by DA DIS, the client will still be required to run the DA DIS process manually.

## Assumptions and Constraints

This file conversion tool will be written in C# using the Microsoft .NET framework 3.5 or higher (4.0 most likely.) This in turn will necessitate the use of Windows XP or higher on client machines. Any processes the tool is to perform will not have any computing power requirements over and above those of the operating system itself so no specific system requirements are necessary. There is a high likelyhood that, in order to facilitate the creation of the DIS-compatible file, any client machine will require software libraries that enable the reading and writing of Microsoft Access 2003 (MDB) or 2007 (ACCDB) files to the extent that these libraries cannot be normally packaged with the final product. Below is a list of specific assumptions:

* The tool will not natively transform just the client’s files, it will require a ‘format definition’ to indicate to the tool which blocks of characters from the input file are to be written to the target table fields.
* The tool will be a Windows form-style application.
* The current DA DIS supports only Microsoft Access files whose format matches that specified in the DA DIS Design Document.
* The table spec in the DA DIS Design Document is up-to-date with the version of RISKMASTER X the client is using.
* The tool will accept existing DIS Access files as generated/updated by the DA DIS ‘DB Tools’ application/module. It will not generate a new file.

## Current Scenario

The client is unable to process their import files as they are not compatible with DA DIS.

## Proposed Scenario

After delivery, the client would process their file in the tool, then take the resulting Access file and import it normally into DIS as described in the DA DIS User’s Guide.

## Functions to be addressed

-Select format definition.

-Select source file.

-Select target file.

-Parse format definition.

-Apply format definition to input file and load data into internal dataset

-Insert data into Access output file.

-Indicate any errors.

# Requirement

In order to meet the needs of the client in converting their input file into a format compatible with DA DIS, this tool needs to meet a certain set of criteria.

Internally, the tool should function by first processing a ‘format definition’ which defines the structure and mapping of the input file to the output Access file. The format definition will provide the tool with information on which tables and fields from the input file need to be mapped to the output Access file so that it may internally generate appropriate data structures. This format definition will initially target only fixed-width files but future compatibility with character-separated files should be kept in mind. The exact specification for this format definition will be covered later or in a future document and must be adequately documented for end-users. No program code will be created for assisting with the generation of this format file. Once internalized, the format definition will be applied to each row of the input file and tokens will be transferred to an internal copy of the Access table and field specified in the format definition. Datatype conversions will be based on the datatype in the Access file and will be automatic, there will be no way to circumvent this conversion. Any errors that result from an improper conversion or a datatype mismatch should halt processing of the file and display a useful error message in an area of the tool screen dedicated to process and error logging. Any errors that occur while parsing the format definition shoud also halt execution and display in the logging area. Once the entire source datafile has been read successfully, the interal data representation will be loaded into the Access target file with appropriate identity fields as specified in the DIS Access file specification. These identity fields can be added into the data set whenever it is most apropos.

The tool should feature a ‘simple’ interface; it has no need for menus, multiple screens or any ‘advanced’ options. File selection dialogs, functional buttons and a log display should be all that is required. The tool should be able to be run in four user-actions: selecting a format file, selecting an input file, selecting an output MDB file and starting the process (button click). To further streamline future executions of the tool, all of the user’s file path selections can be persisted between sessions.

There are no specific performance requirements for this tool.

## Functional Considerations

While the format will allow an end user to specify nonsense file formats (ie. Combining some arbitrary admin tracking table with payments with vehicles with org hierarchy) there is no requirement that the tool validate that formats make sense or only cover a single import area. It is left as an exercise of the user to know and understand the type of data they are bringing into the system. That being said, the format specification should implement an ‘OMIT’ or ‘NULL’ mapping for a field from a input file so that a single file could be processed with multiple formats into multiple tables. For instance, an employee list of cab drivers with vehicle details about their cabs could be processed with a format to populate employee data and then again to process vehicle data.

As stated above, the tool should indicate to the user if there was a problem with parsing the format description. The input file should be nominally checked against the format description only with regards to the length of a row. Lines of the input file which are too short should cause a halt action and an error message should indicate the offending line number. Lines which are too long in the input file should either show up as warnings or be ignored; again, the user should be familiar with their data but at the same time it should not be a requirement to map all the fields in a file when you only need the first *x.* The target should be given a cursory check that it is, in fact, an Access file and that the fields specified in the format definition do exist in its tables.

Any error previously with a ‘halt’ action should prevent any data being loaded into the Access file. No partial loads.

How the tool should react when it receives an Access file which is not empty is yet to be determined. The probable solution is that the tool could either clear the tables before loading or halt with an error stating that an empty Access file (for at least the tables specified in the format definition) must be used.

The creation of file versions of any error/warning/process messages is [not required/optional/mandatory] (you decide!)

## Design Considerations

If any Access access libraries do not support a feature set required for this application, there is a possibility that an ODBC connection to an Access file could be supplied instead of the actual path to an Access file. All possible efforts should be made to utilize current methods of access (ADO.NET.)

## Interface Considerations

[The requirement states how the software should interact with people, the system's hardware, other hardware, and other software. This should be a detailed description of all inputs and outputs from the system or process.]

### System Interfaces

No new system interfaces or changes to existing interfaces are needed for this requirement.

### Hardware Interfaces

No new hardware interfaces or changes to existing hardware interfaces are needed for this requirement.

### Software Interfaces

No new software interfaces or changes to existing interfaces are needed for this requirement.

### User Interfaces

The tool will provide a simple, straightforward user interface featuring file select dialogs and a modest space for message/error/etc. output. Settings in each user input field should be persisted between executions of the program on a particular computer (Settings/config serialization, not registry, not written to database or other storage.)

## Performance Considerations

Functional delivery exit criteria do not include performance testing. Performance testing requirements will be defined, where applicable, as part of the RISKMASTER GA release Performance test plan.

# Dependencies between Requirements

Because there is only one requirement that comprises this functional delivery, there are by definition no dependencies on other requirements.

# Approach

By creating this tool for converting the client’s input file to an Access file which can be accepted by the DA DIS template, we hope to add only a single, unobtrusive step into the clients processing of such input files without necessitating significantly more involved changes to both the DA DIS template and the RMX DIS optionset interface pages both of which would require separate documentation, development and testing.

# Signoff

**The signatures below represent approval and signoff of the requirements and approaches as listed above.**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Reviewer(s) Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Testing Lead Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Program Manager Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Development Manager Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Hosting Manager Date

Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# References

1. *DA DIS Design*
2. *DA DIS Format Specification (may be the same document as #1)*
3. *DA DIS User’s Guide*
4. *Import File Layout*
5. *Thomas McGee DIS Conversion Tool Design Specification*

# Requirement Change Log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Change # | Requirement # Revised | Nature of Revision | Initials |
| 1/27 | 0 |  | Original Version | BS |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |