

Export - Import Feature

See below for import/export format and usage notes.

Export function

For export, from the File Menu select “Export Receivers to...” and browse to the location you wish to create the export file.

See below for export file format.

Import function

For import from data generated from other applications:

Please note that for import the first 5 fields are required (tab-delimited):

ReceiverName, ReceiverID, MessagingService, LoggingFlag, and RouteOnFailureFlag

Example: John Doe<tab>7891234567<tab>Paging Service Name<tab>1<tab>0

ALSO, the MessagingService name must exist in the Messaging Service database before import.

From Excel or database program save in tab delimited format. Please note that some versions of Excel save files in UTF-8 or Unicode encoded format by default, you may then need to open the text file in NotePad and save as “ANSI” encoded format before importing.

From the File Menu select “Import Receivers from..” and browse to your tab-delimited import file.

Export / Import file specification

ReceiverName<tab>ReceiverID<tab>MessagingService<tab>LoggingFlag<tab>RouteOnFailureFlag<tab>AltReceiverID<tab>AltMessagingService[<tab>GroupName1<tab>GroupName2<tab>Groupname3<tab>...]

Field Name	Content Type	Required?	Length (in characters)	Comments
ReceiverName	Alphanumeric	yes	32 max.	
ReceiverID	Alphanumeric	yes	256 max.	
MessagingService	Alphanumeric	yes	32 max.	
LoggingFlag	Numeric	yes	1	0 – message logging disabled 1 – message logging enabled
RouteOnFailureFlag	Numeric	yes	1	0 – route on failure disabled 1 – route on failure enabled
AltReceiverID	Alphanumeric	yes	256 max.	Ignored if RouteOnFailure flag is 0
AltMessagingService	Alphanumeric	yes	32 max.	Ignored if RouteOnFailure flag is 0
GroupName	Alphanumeric	optional	32 max.	

Export / Import Notes

- The file consists of ASCII text, tab-delimited, one receiver entry per line (ANSI format)
- User is informed that importing a text file will erase all current receiver and group info, regardless of the result of the import operation
- On export the user selects the output file name and location with a File Save dialog
- On import user selects the input file with a File Open dialog
- On import file is processed line by line, with any problem causing the line to be skipped. Possible problems include:
 - Any of the required fields missing
 - Any field containing invalid information
 - Any field exceeding its maximum size
 - Line exceeding any OS-dependent limit on line length
 - Number of optional groups exceeding product group limit
- If there is an error during import a pop-up will be displayed showing the line number where the error occurred and any other info that we can include that will help in fixing the problem.
- The Services database (Services.irt) must already exist, and the MessagingService and/or AltMessagingService must already exist in the Services.irt database for import.
- ReceiverName, ReceiverID, MessagingService, LoggingFlag, and RouteOnFailureFlag are required fields for import
- RouteOnFailure feature available in AlphaPage First Responder, Watch-IT and Enterprise products

Export / Import Usage notes

- The program will export to a tab delimited text file (ANSI format) which can then be imported into an Excel spread sheet or database as desired.
- From Excel or database program save in tab delimited format. Please note that some versions of Excel save files in UTF-8 or Unicode encoded format by default, you may then need to open the text file in NotePad and save as "ANSI" encoded format before importing.
- If not using RouteOnFailure, but are using Groups, note there will be 3 tabs between RouteOnFailureFlag and the first Groupname.

To illustrate, if using RouteOnFailure the data after the RouteOnFailureFlag would be:

```
<tab> AltReceiverID <tab> AltMessagingService <tab> FirstGroup  
(note the 3 tabs from RouteOnFailureFlag to the first group name)
```

The 'tab' is required "between" fields. If not using RouteOnFailure, then the 2 empty fields cause there to be 3 consecutive tabs.

Therefore, if not using RouteOnFailure the data after RouteOnFailureFlag would be:

```
<tab> <tab> <tab> FirstGroup
```