## Mail Merge

Mail merge refers to an operation by which WordprocessingML documents may work in conjunction with data from an external data source, importing this data into a document according to a set of codes contained in WordprocessingML known as fields.

A WordprocessingML document that contains the [mailMerge](mailMerge.docx) element (§) and is therefore connected to an external data source, is known as a source document. In addition to being connected to an external data source and containing fields, a source document may contain any regular WordprocessingML constructs such as:

* Text runs
* [Paragraphs](Paragraphs.docx)
* Images
* [Tables](Tables.docx)
* [Numbering](Numbering.docx)
* Etc.

There are two key parts of the mail merge data stored in a WordprocessingML document

1. Information connecting a document to an external data source
2. Information populating fields within that document with external data.

Once the fields in a merged document have been populated with external data, mail merge has been completed and the resulting files are known as mail merged documents or simply merged documents.

The mail merge settings for a WordprocessingML document are stored in two locations:

* The standard mail merge settings are stored as the child elements of the [mailMerge](mailMerge.docx) element (§)
* A set of additional mail merge settings stored in the [odso](odso.docx) element (§), and collectively referred to as the Office Data Source Object settings. The Office Data Source Object is an extension to the standard settings stored with a mail merge which performs two functions: First, it provides additional information about the mail merge data source, specifically: information about how to map the columns in the data source to [MERGEFIELD](MERGEFIELD.docx) fields and information about records which shall be included and excluded when creating merged documents. Second, it provides an alternate set of connection information which should be used when the [dataType](dataType.docx) element (§) specifies a value of native. This alternate connection string provides additional connection information for applications which choose to support the ODSO connection string syntax.

[Example: Consider a WordprocessingML document containing static WordprocessingML constructs such as text runs and paragraphs in addition to two WordprocessingML [MERGEFIELD](MERGEFIELD.docx) fields (§) calling for Courtesy Title and Last Name data. The field codes for each field are displayed, delimited by {} characters:

|  |
| --- |
| Dear {MERGEFIELD "Courtesy Title" \m} {MERGEFIELD "Last Name" \m},  Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text. Sample text.  Sincerely, |

If the following WordprocessingML was added to this document, this document would become a source document rather than just a standard WordprocessingML document, as the [mailMerge](mailMerge.docx) (§) element specifies the elements and attributes necessary to [enabled](enabled.docx) the document to connect to an external spreadsheet data source.

<w:[mailMerge](mailMerge.docx)>

…

<w:[dataType](dataType.docx) w:val="spreadsheet" />

<w:[query](query.docx) w:val="SELECT \* FROM `Sheet1$`" />

<w:[dataSource](dataSource.docx) r:id="rId1" />

…

</w:[mailMerge](mailMerge.docx)>

Here, the [dataType](dataType.docx) (§) and [dataSource](dataSource.docx) (§) elements specify that the given document shall be connected to the external data source target by the relationship whose relationship value is rId1 as specified in the [dataSource](dataSource.docx) element (§). While connected to the external data source, the source document together with the hosting application and/or data source access application will extract data from the external data source to perform the merge as specified by the [connectString](connectString.docx) (§) and [query](query.docx) (§) elements. end example]