How to diagnose issues with Transact NoFileEnquiry Interface?

Ivana Dordevic - 2022-06-22 - Comments (0) - Temenos FAQs

temenos

Interface Overview

The No File interface adds the ability to return a list of documents that are securely stored within the Autoform DM Archive through a Transact enquiry. The list of returned documents can be limited by passing search criteria to DM. For example, the Customer Number could be passed to retrieve a list of archived documents for a specific customer.

This article is only relevant to the older Transact installations.

How does the connection work?

The interface is designed to be run using a specially created Transact Enquiry, where the DM search criteria and the search name are specified. Transact connects to DM over a SOAP web service call when a No File enquiry is run. This passes the relevant information to DM so that it is able to correctly interpret what search it should run, the search criteria and what values should be returned to be displayed on the Transact enquiry screen. The Integration manual, which is supplied during the initial installation, contains all of the interface configuration information as well as what files are necessary. The debugging instructions below are based on a JBOSS Temenos install, however the functionality and any possible issues should be broadly similar for other platforms (though the file locations may vary).

Logging

The logging of the No File interface is configured during the initial installation and this documentation can be found in the Integration manual. By default, the logging of the interface will appear in the JBoss log at the Debug level. It is important to remember that as this interface interacts with DM, you can also find useful information for debugging in the DM logs as well. There are additional articles on the Support website that detail how to turn on Debug mode, as well as finding and reading the DM logs. When a No File enquiry is run with Debug logging turned on, the JBoss log should show messages similar to below:

NoFile JBoss Logging

```
DEBUG [com.efstech.temenos.nofile.NoFileEnquiry] (WorkManager(2)-25)
```

```
Provided mapAsString from T24: SEARCH='ByDocumentID';Customer
Number=1,document.name=2,document.id=3
DEBUG
[com.efstech.temenos.nofile.com.efstech.temenos.common.PropertyBasedC
onfiguration] (WorkManager(2)-25) Attempting to initialise 'class
com.efstech.temenos.nofile.NoFileEnquiry' using
'/NoFileEnquiry.properties'
DEBUG
[com.efstech.temenos.nofile.com.efstech.temenos.common.PropertyBasedC
onfiguration] (WorkManager(2)-25) Attempting to load properties using
prefix 'null'
DEBUG
[com.efstech.temenos.nofile.com.efstech.temenos.common.PropertyBasedC
onfiguration] (WorkManager(2)-25) Attempting to load property [URL
(dm.url)] using prefix [null]
DEBUG
[com.efstech.temenos.nofile.com.efstech.temenos.common.PropertyBasedC
onfiguration] (WorkManager(2)-25) Got value
[http://127.0.0.1:80/pdm/servlet/NoFileServlet] for property [URL
(dm.url)]
```

This is a highlight of some of the logging that will appear in the Transact JBoss logs. The first line in this example shows that the DM search to be run is called 'ByDocumentID' and then the customer number, document name and document id are the results to be returned. Below is a highlight of the relevant logging that will appear in the DM logs when a No File enquiry is run. Here you can see the user logging into DM and then further down the search, 'ByDocumentID', being run. The keys to return are also specified here.

NoFile DM Logging

```
- [efstech.pdm.web.servlet.NoFileServlet                  ] INFO  -
NoFileServlet request with params:
{params=[Ljava.lang.String;@c210d1}
- [pdm.securityservice.jaas.jboss.JBossLoginModule        ] INFO  -
Getting login module
- [efstech.pdm.configurationservice.util.LDAPSettings     ] DEBUG -
Get LDAPEnabled: false
- [pdm.securityservice.jaas.jboss.LoginModuleFactory      ] DEBUG -
Instantiating DB implementation
- [pdm.userservice.ejb.session.UserAdminSessionFacadeEJB  ] DEBUG -
Getting UserEntity for username: admin
- [efstech.pdm.jpa.dao.UserDaoJPA                         ] DEBUG -
Getting UserEntity with username: admin
```

- [efstech.pdm.jpa.dao.UserDaoJPA] DEBUG - Found UserEntity: UserEntity{id=1, username='admin', type=2, accountExpiry=null, emailAddress='', passwordNoExpire=true, lastLogin=2016-04-01 12:50:26.39, incorrectLoginAttempts=0, bypassLockout=true, changePasswordNextLogin=false}
- [efstech.pdm.jpa.entities.UserEntity] DEBUG - User[admin] Latest password: PasswordEntity{userId=1, password='e3274be5c857fb42ab72d786e281b4b8', lastChanged=2015-08-11 16:03:52.988}
- [pdm.securityservice.jaas.jboss.DBLoginModuleImpl] DEBUG - Attempting standard PDM login for admin
- [pdm.securityservice.jaas.jboss.AbstractLoginModule] DEBUG - Authentication of [admin] successful
- [pdm.userservice.ejb.session.RoleQueryServiceEJB] DEBUG - Retrieved Groups from cache for user: admin
- [pdm.securityservice.jaas.jboss.AbstractLoginModule] DEBUG - Got role sets for user [admin]: [Roles(members:DeleteDocDefExtendedKeyDef,CreateApplication,CreateExtendedUserProperty,ExecuteServerScript,CreateSchedule,DisableUser,CreatePrintApplication,DeleteExtendedDataDef,DeleteDocDefKeyBinding,DeleteGroup,ChangeUserPassword,pdm,CreateDocDef,UpdateGroupMembership,DeleteSchedule,UpdateUserRoles,CreateBatchDefinition,UpdateSchedule,DeleteNoteDefinition,CreateExtendedDataDef,ExecuteApplication,ManageV4Migration,UpdateArchivePath,UpdatePasswordSettings,UpdateDocumentSettings,UpdateGroupApplicationAccess,UpdateServerSettings,UpdateMisc,UpdateDocDefKeyBinding,CreateDocDefExtendedKeyDef,DeleteRole,ChangeOwnPassword,UpdateDocDefName,UpdateNoteDefinition,DeleteRecordLock,CreateDocDefKeyBinding,UpdateRoleMembership,UpdateLoggingSettings,UpdateLdapGroupMappings,UpdatePrintApplication,UpdateDocDefExtendedKeyDef,UpdateEmailSettings,UpdateBatchDefinition,CreateSubGroup,CreateRole,UpdateServerScript,UpdateAddressBookEntry,UpdateApplication,ReadDocumentStatistics,UpdateRoleName,ReadLdapSettings,CreateUser,pdmAdmin,UpdateExtendedUserPropertyValue,DeleteAddressBookEntry,CreateNoteDefinition,CreateAddressBookEntry,UpdateAuditLogSettings,CreateArchivePath,DeleteUser,UpdateLdapSettings,UpdateExtendedDataDef,DeleteApplication,ReadServerStatus,CreateDocDefExtendedKeyBinding,UpdateGroupName,UpdatePrintSettings,DeleteArchivePath,ReadAddressBookEntry,CreateGroup,DeletePrintApplication,UpdateApplicationAccess,DeleteDocDefExtendedKeyBinding,DeleteSessionLock)]
- [efstech.pdm.web.temenos.TemenosSearchServiceEJB] DEBUG - Decoded params: SEARCH='ByDocumentID';Customer

```
Number=1,document.name=2,document.id=3
- [efstech.pdm.web.temenos.TemenosSearchServiceEJB         ] DEBUG -
Got return keys: [document.name, Customer Number, document.id]
- [efstech.pdm.searchservice.facade.SearchServiceFacadeEJB ] DEBUG -
Getting documents with search: ByDocumentID for page: 1 and criteria:
[]
- [pdm.searchservice.ejb.session.SearchSessionFacadeEJB    ] DEBUG -
Getting SearchEntity with name: 'ByDocumentID'
```

Further down the DM log files, the exact SQL query that is run against the DM SQL database is viewable. In this example, 11 results have been found which will then subsequently return the document name, customer number and document id to the enquiry screen in Transact.

```
- [com.efstech.pdm.searchservice.SearchDAO                 ] DEBUG -
About to execute SQL: SELECT DISTINCT tblDocuments.docID,
tblDocuments.docName, tblDocuments.fileMD5Hash,
tblDocuments.fileVersion, tblUsers.username,
tblDocuments.lastChanged, tblDocuments.fileArchived,
tblDocuments.revision , tblDocuments.SKEY1, tblDocuments.docName AS
Expr1, tblDocuments.docID AS Expr2 FROM tblDocuments INNER JOIN
tblUsers ON tblDocuments.ownerID = tblUsers.id  WHERE
tblDocuments.docTypeID = 42949672989 AND tblDocuments.isDeleted = 0
ORDER BY fileArchived DESC
- [com.efstech.pdm.searchservice.SearchDAO                 ] DEBUG -
Found 11 results
- [pdm.loggingservice.ejb.session.LogServiceEJB            ] DEBUG -
Potentially logging action SEARCH_RUN, with parameters:
[ByDocumentID( 180388626435 ), SELECT DISTINCT tblDocuments.docID,
tblDocuments.docName, tblDocuments.fileMD5Hash,
tblDocuments.fileVersion, tblUsers.username,
tblDocuments.lastChanged, tblDocuments.fileArchived,
tblDocuments.revision , tblDocuments.SKEY1, tblDocuments.docName AS
Expr1, tblDocuments.docID AS Expr2 FROM tblDocuments INNER JOIN
tblUsers ON tblDocuments.ownerID = tblUsers.id  WHERE
tblDocuments.docTypeID = 42949672989 AND tblDocuments.isDeleted = 0
ORDER BY fileArchived DESC, []]
- [pdm.searchservice.ejb.session.SearchSessionFacadeEJB    ] DEBUG -
Got search results: [SearchResultsDto{results=[210453397539,
AccountStatement, 2EFFF0608F55A9A67D26BCDEA99B811A, 1, admin,
```

5/1/2016 12:57, 5/1/2016 12:57, 1, 100336, AccountStatement,
210453397539]}, SearchResultsDto{results=[141733920923,
AccountStatement, 3C6F45DDE1B7CD6BA1851AAF09F0D61A, 1, admin,
2/9/2016 15:33, 2/9/2016 15:33, 1, 100336, AccountStatement,
141733920923]}, SearchResultsDto{results=[141733920791,
AccountStatement, 1C168F8BCF9729A552CC074D61395DD4, 1, admin,
2/9/2016 10:08, 2/9/2016 10:08, 1, 100336, AccountStatement,
141733920791]}, SearchResultsDto{results=[98784247958,
AccountStatement, A346096DBA69D1CBE990D6CA7E120858, 1, admin,
11/3/2015 16:21, 11/3/2015 16:21, 1, 100409, AccountStatement,
98784247958]}, SearchResultsDto{results=[98784247954,
AccountStatement, F8750BC030F3BBEDE70F137F8C9977BC, 1, admin,
11/3/2015 16:16, 11/3/2015 16:16, 1, 100409, AccountStatement,
98784247954]}, SearchResultsDto{results=[98784247950,
AccountStatement, 12419CBC112960F9296CC545AB6866D9, 1, admin,
11/3/2015 16:13, 11/3/2015 16:13, 1, 100409, AccountStatement,
98784247950]}, SearchResultsDto{results=[98784247949,
AccountStatement, CF540FBDF833151E20E6DD9857C72156, 1, admin,
11/3/2015 16:13, 11/3/2015 16:13, 1, 100409, AccountStatement,
98784247949]}, SearchResultsDto{results=[98784247813,
AccountStatement, D08670F5BC437179D4622A6F968B17C3, 1, admin,
11/3/2015 15:05, 11/3/2015 15:05, 1, 100409, AccountStatement,
98784247813]}, SearchResultsDto{results=[60129542282,
AccountStatement, CDEEF2F4A9B1917720C77FFEA37BABAD, 1, admin,
10/6/2015 15:57, 9/10/2015 11:02, 5, 100336, AccountStatement,
60129542282]}, SearchResultsDto{results=[60129542281,
AccountStatement, 300A58E9B1CD4B3694E8E21013DDB6C6, 1, admin,
10/6/2015 15:57, 9/10/2015 11:01, 4, 100336, AccountStatement,
60129542281]}, SearchResultsDto{results=[42949673012,
AccountStatement, 75663BAE261892C449D78816D89312FA, 1, admin,
10/6/2015 15:57, 8/21/2015 15:59, 5, 100336, AccountStatement,
42949673012]}]

When reporting interface issues to Formpipe Support, please have the Transact and DM logs for the time of the error and ensure the system is in Debug mode already.

Possible Issues

Most issues are generally caused by network interference or incorrect configuration. Basic troubleshooting steps for diagnosing common No File issues are described below. However, due to the complexity of the interface, it is impossible to cover every scenario. If you experience any difficulty in troubleshooting the interface please contact Formpipe

Support and we will be happy to discuss and assist with any issues that you may encounter.

Configuration

No File relies heavily on a stable network connection between the Transact and DM servers. If there is any interference between the two, this could cause issues with the No File functionality. The main configuration file is NoFileEnquiry.properties which is located within the efs-nofileenquiry-4.1.3.jar (the numbers will change depending on your interface version). The jar file can be opened with an archiving program such as 7Zip or WinRar to access the file.

NoFileEnquiry.properties

```
dm.url=http://127.0.0.1:80/pdm/servlet/NoFileServlet
dm.username=admin
dm.password=password
dm.retries=3
dm.wait.time=3
#dm.password=ENC(encrypted_password)
#dm.password.encryptionconfig=Pbkdf2HmacSha1WithAes128Cbc
```

The Transact enquiry passes the search parameters to DM using a web service call over the port defined in the file above. The results are then passed back to Transact along the same route.

How to test the network

In order for the interface to function correctly, the DM server needs to be reachable by the Transact server and that the port used is open across all relevant firewalls. All connections to the Autoform DM server will be made directly from the Transact server and not from any client computers. Network connections can be tested using the following methods. ol>

- Ping the DM server from the Transact server. This should resolve the DM server name to an IP address and return a reply with an appropriate TTL time.
    - If the IP address fails to resolve, the issue is likely to be with the server name or DNS.
    - If the IP address resolves but the server is not found, this is likely to indicate a problem with the connection between the Transact and DM server.
- Telnet to the DM server using the defined port from the Transact server. This should result in a blinking cursor on a blank screen.
    - If this returns an error or times out, then the port on the DM server is not reachable. This is most likely due to a firewall blocking connections to that port.
- Access the DM web interface from the Transact server a web browser: http://dmServerName:port/dm
    - If DM is accessible and running, then the web interface will load and you will

be able to log in.
    - If the webpage cannot be found then it is either not accessible or the DM service is not running.

Further connection checks

Using the DM NoFile servlet URL from the NoFileEnquiry.properties file, (http://127.0.0.1:80/pdm/servlet/NoFileServlet - taken from the above example) there is another check that can be done. By accessing the URL through a web browser, an HTTP 500 error message should appear showing that an error has occurred due to there being no username in the parameters. The presence of this error shows that DM is accessible and running from the current computer. If the HTTP 500 error message does not appear and instead you receive a "You're not connected to a network" message, then double-check that the Transact and DM server are connected and can communicate. If everything seems to be working correctly and there is still no connection, one final check can be made by taking the encoded servlet string out from the Transact logs and running it in a web browser. Example below:

```
http://127.0.0.1/pdm/servlet/NoFileServlet?username=admin&password=password&params=5345415243483d274279446f63756d656e744944273b437573746f6d6572204e756d6265723d312c646f63756d656e742e6e616d653d322c646f63756d656e742e69643d33
```

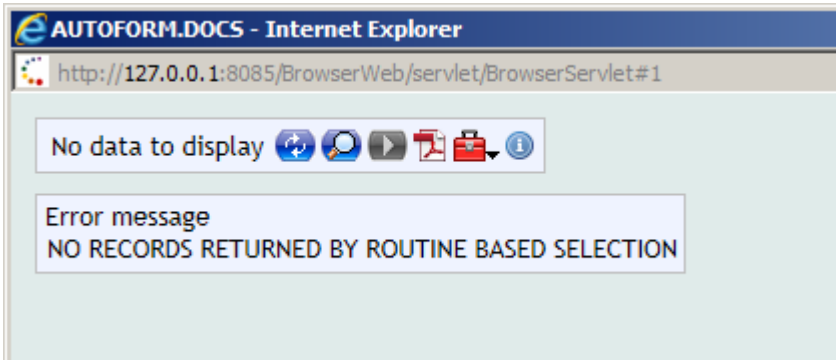
Connecting to the relevant URL (in the Transact logs) from a web browser, passes the parameters to DM, which are then decoded and run. This will return the results as a string. If this is run from the Transact server in a web browser, you can be certain that DM is accessible and that it is trying to pass information back to the enquiry. The example above returns the string below (customer number, document name and document id) when it is run.

```
100336@VMAccountStatement@VM210453397539@FM
100336@VMAccountStatement@VM141733920923@FM
100336@VMAccountStatement@VM141733920791@FM
100409@VMAccountStatement@VM98784247958@FM
100409@VMAccountStatement@VM98784247954@FM
100409@VMAccountStatement@VM98784247950@FM
100409@VMAccountStatement@VM98784247949
```

Incorrect search name

The Autoform DM search name that is defined in the Transact enquiry must match the

name that has been set for the search in DM. If they are different then DM will not be able to find the search that has been specified and will generate an error. In the Transact enquiry screen, an error message will appear (as shown below). To find out why this error has occurred you must look in the log files.



The Transact logs will display an 'IllegalStateException' stack trace, indicating that it did not get a success message from the server. As a consequence, a few lines of HTTP code will be displayed above the beginning of the stack trace. This is returned from DM and although it is not displayed anywhere, it does contain useful information for the administrator. Within this code will be a message indicating that no such search was found for the given name. In the DM logs another error will be displayed, also indicating that there is no such search for the given name. To fix this problem, open the Transact enquiry design screen and log into the DM web interfaces at the same time to compare the search name used. We recommend changing the Transact enquiry search name only, as the DM search may be used in other applications and enquiries, which could break if the name is changed. Transact may need to be restarted and the browser cache cleared, for the updated search name to be used.

Not getting the expected results?

If the results that are being returned to the enquiry are not what is expected, then running the relevant search in the DM web interface with the parameters that are captured in the Transact logs will allow you to easily test if the search is working correctly. To find the actual parameters that are being passed to DM, look in the Transact logs and locate the line that is similar to the one below.

```
DEBUG [com.efstech.temenos.nofile.NoFileEnquiry] (WorkManager(2)-29)
Provided mapAsString from T24:
document.id=1234595685,SEARCH='ByDocumentID';Customer
Number=1,document.name=2,document.id=3
```

This line will only appear in the logs if they are in DEBUG mode. The line can be broken down to show what is happening.

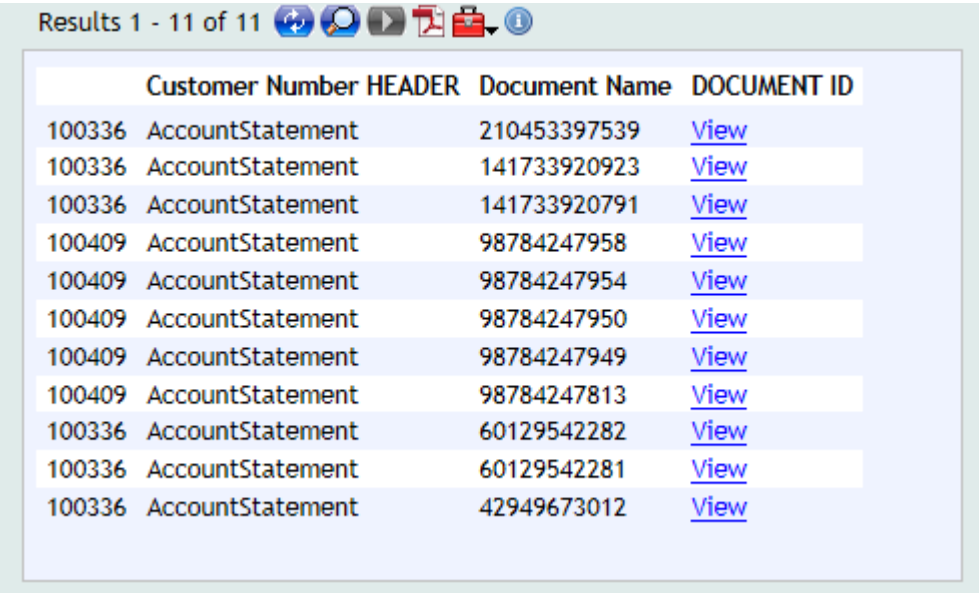
- document.id=100336
    - Document ID is the search parameter and 100336 is the value that is being passed to it.

- SEARCH='ByDocumentID'
    - This is the search that has been created in DM and is going to be run with the passed parameter.
- Customer Number=1, document.name=2, document.id=3
    - These are the results that will be returned to the Transact enquiry, they have been created in both DM and the Transact enquiry.

By taking out the search parameters and search name, you can then execute the search in the DM web interface to double check that it is working correctly and the results you expect to appear are returning.

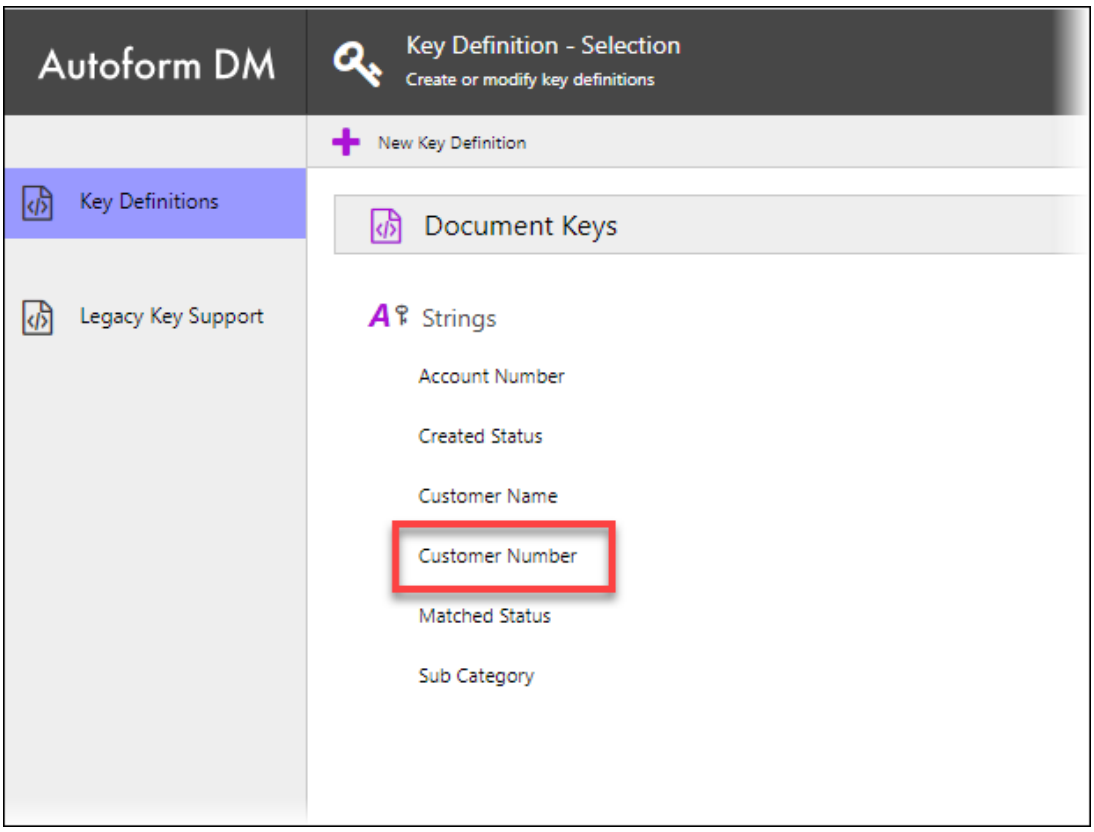
Autoform DM key names and Transact field names

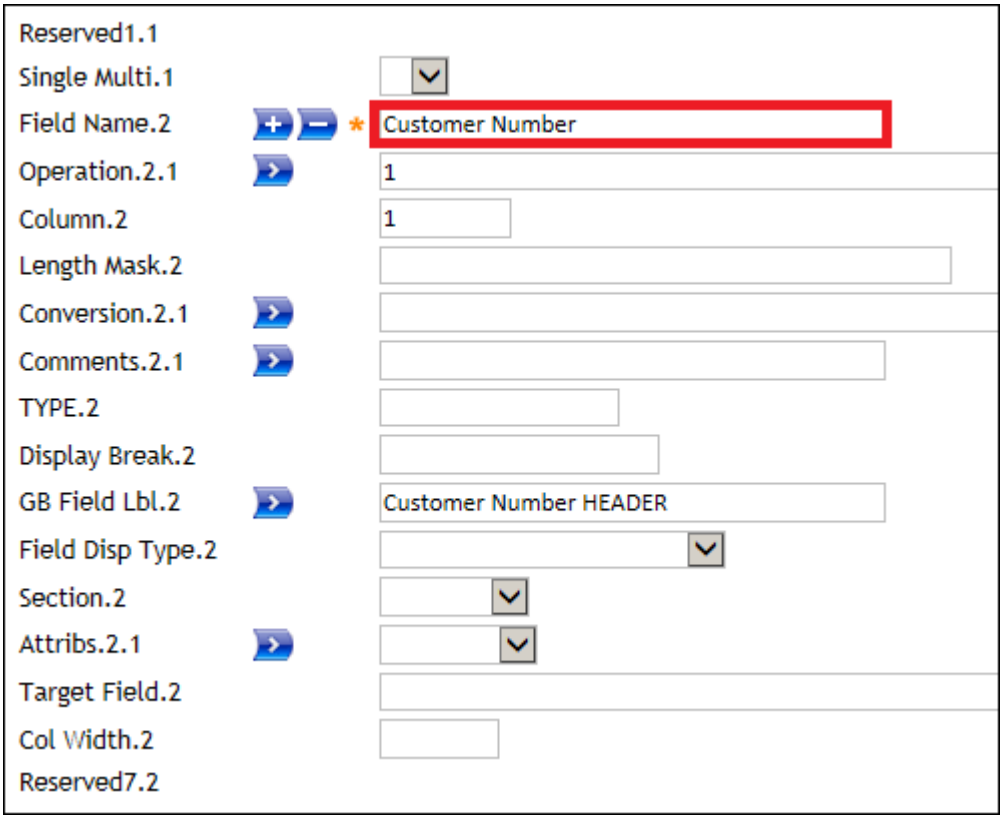
When designing the Transact enquiry and Autoform DM search, it is important that the DM key names and Transact field names match, so that the returned data is correctly displayed. If the names do not match, then the results will likely be displayed as in the example below. In this example, the first column Field Name should be 'Customer Number', but it has been set incorrectly and does not match what is used in DM. Therefore Transact does not know what column header to set as it does not match anything in the enquiry design, so a blank header is used and the next headers are pushed along the columns.


Results 1 - 11 of 11

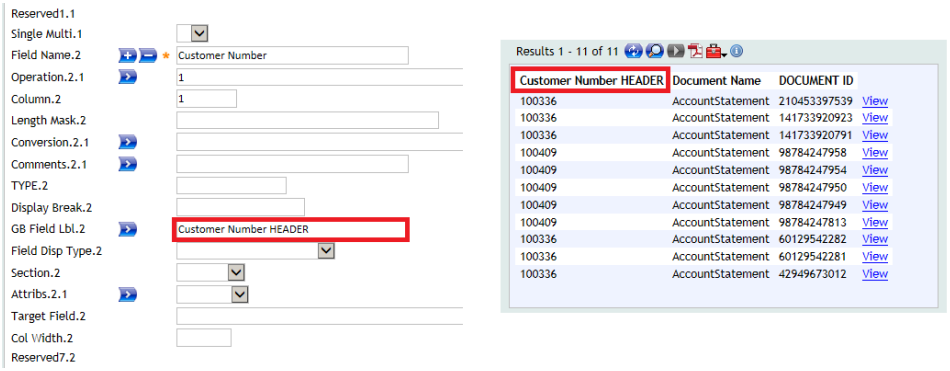
	Customer Number HEADER	Document Name	DOCUMENT ID
100336	AccountStatement	210453397539	View
100336	AccountStatement	141733920923	View
100336	AccountStatement	141733920791	View
100409	AccountStatement	98784247958	View
100409	AccountStatement	98784247954	View
100409	AccountStatement	98784247950	View
100409	AccountStatement	98784247949	View
100409	AccountStatement	98784247813	View
100336	AccountStatement	60129542282	View
100336	AccountStatement	60129542281	View
100336	AccountStatement	42949673012	View

In Transact enquiry design, the Field Name has to match the DM key name:





The Field Label will be displayed in the enquiry as the column header:



If the Field Names match in both the Transact enquiry and DM, then the results will be returned and displayed correctly.