

Predator - How to Setup Link Analysis

Use this how-to guide to setup Link Analysis

The Link Analysis page is available from within the Transaction Review page as a link next to the field that has that ability enabled. It displays the results of a SQL Report in a visual fashion, highlighting links between property fields, e.g.: accounts, customers, etc.

The purpose of Link Analysis is to allow AML/Fraud investigators to visually analyse what Property Fields are linked to each other thus ease fields relationship detection.

Link Analysis Setup

1. go to Maintenance > Link Analysis Maintenance
2. You will need to define a pair of link for each link analysis
 - a. From account To account
 - b. To account From Account
3. The following are the definition of the
 - a. Link Analysis ID - Unique ID which identifies the link. Allow integer only
 - b. From Property Id - The property field that define the starts of the link
 - c. To Property Id - The property field that define the end of the link
 - d. Graph Query
 - i. The query that link the from property Id to To Property Id
 - ii. This query has two parameter, which are pass when you navigate to Link Analysis Screen. @ID = Sender Account No (From Property ID) and @AmountThreshold = Amount Threshold.

Link Analysis

Sender Account No	From	To	Amount Threshold	Analyse
ACT001	29/08/2019 17:22	29/08/2019 17:22		

- e. Table Query - Query that build the table when you select the connector. This query has 5 parameter
 - i. @FromId - the fields that is start of the link
 - ii. @ToId - the fields that is end of the link
 - iii. @AmountThreshold - The amount threshold in link analysis screen
 - iv. @SortField - If define, when you click on the Sort while viewing the connector, the field will be sorted

ACT001

ACT003

Serial No	Group By	Date &Time	Amount	Channel	Currency Code	Transaction Code
5	164421091120112	10/08/2018 12:00:00 PM	888.00	Cards C02	AUD	101
2	164421091120112	10/08/2018 12:00:00 PM	1,000.00	Internet Banking C01	AUD	101

Showing 1 to 2 of 2 records.

1

- v. @SortOrder - Valid Value will be 0 (ascending) and 1 (descending).
- f. Additional Column 1 - First additional column that you would like to populate in the table. If define, it will be visible when you click on the connectors

ACT001

ACT003

Serial No	Group By	Date &Time	Amount	Channel	Currency Code	Transaction Code
5	164421091120112	10/08/2018 12:00:00 PM	888.00	Cards C02	AUD	101
2	164421091120112	10/08/2018 12:00:00 PM	1,000.00	Internet Banking C01	AUD	101

Showing 1 to 2 of 2 records.

1

- g. Additional Column 2 - Second additional column that you would like to populate in the table. If defined, it will be visible when you click on the connectors

ACT001				ACT003			
Serial No	Group By	Date & Time	Amount	Channel	Currency Code	Transaction Code	
5	164421091120112	10/08/2018 12:00:00 PM	888.00	Cards C02	AUD	101	
2	164421091120112	10/08/2018 12:00:00 PM	1,000.00	Internet Banking C01	AUD	101	

Showing 1 to 2 of 2 records.

Example of a Link Analysis

In this example we will create a link to trace transaction amount moving from an account to another account. Below is the sample scenario of transaction moving from 3 different accounts.

Channel	From Account	To Account	Transaction Amount
C01	ACT001	ACT003	1000
C01	ACT001	ACT002	1500
C01	ACT002	ACT003	2500
C02	ACT001	ACT003	888
C02	ACT003	ACT002	950
C02	ACT002	ACT001	760
C02	ACT002	ACT001	760

To configure the link analysis

1. From Account To Account
 - a. Details

Details

Link Analysis Id: 1

From Property Id: Sender Account No

To Property Id: Receiver Account No

Graph Query:

```
DECLARE @report TABLE (
    Transaction_Serial_No BIGINT
    Primary Key,
    Transaction_Datetime DATETIME,
```

Table Query:

```
SELECT * FROM (SELECT
    Transaction_Serial_No as
    SerialNumber,
    Transaction_Datetime as
```

Additional Column 1: ☒ Currency Code

Additional Column 2: ☒ Transaction Code

Save Delete Cancel

- b. Graph Query

Graph Query

```
DECLARE @report TABLE (
    Transaction_Serial_No BIGINT Primary Key,
    Transaction_Datetime DATETIME,
```

```

Transaction_Amount DECIMAL(19,2),
FromId NVARCHAR(25),
ToId NVARCHAR(25)
)

INSERT INTO @report (
Transaction_Serial_No,
Transaction_Datetime,
Transaction_Amount,
FromId,
ToId
)
SELECT
Transaction_Serial_No,
Transaction_Datetime,
Transaction_Amount,
Sender_Account_No,
Reciever_Account_No
FROM [dbo].[C01_Details] a (NOLOCK)
WHERE
(Sender_Account_No = @Id) AND
(Transaction_Amount >= @AmountThreshold OR
@AmountThreshold IS NULL) AND
Transaction_Datetime >= @From AND
Transaction_Datetime <= @To

UNION
SELECT
Transaction_Serial_No,
Transaction_Datetime,
Transaction_Amount,
Sender_Account_No,
Receive_Account_no
FROM [dbo].[C02_Details] a (NOLOCK)
WHERE
(Sender_Account_No = @Id) AND
(Transaction_Amount >= @AmountThreshold OR
@AmountThreshold IS NULL) AND
Transaction_Datetime >= @From AND
Transaction_Datetime <= @To

SELECT
FromId,
ToId,
SUM(Transaction_Amount) AS [Total],
COUNT(*) as [Count]
FROM @report
GROUP BY FromId, ToId

```

c. Table Query

Table Query

```
SELECT * FROM (SELECT
    Transaction_Serial_No as SerialNumber,
    Transaction_Datetime as TransactionDateTime,
    Transaction_Amount as TransactionAmount,
    Card_No as GroupBy,
    'Source Account No' as GroupByColumnName,
    'Internet Banking C01' as ChannelId,
    Currency_Code as AdditionalColumn1,
    Transaction_Code as AdditionalColumn2
FROM [dbo].[C01_Details] a (NOLOCK)
WHERE
    (Sender_Account_No = @FromId) AND
    (Reciever_Account_No = @ToId) AND
    (Transaction_Amount >= @AmountThreshold OR
@AmountThreshold IS NULL)
UNION
SELECT
    Transaction_Serial_No as SerialNumber,
    Transaction_Datetime as TransactionDateTime,
    Transaction_Amount as TransactionAmount,
    Account_No as GroupBy,
    'Source Account Number' as GroupByColumnName,
    'Cards C02' as ChannelId,
    Currency_Code as AdditionalColumn1,
    Transaction_Code as AdditionalColumn2
FROM [dbo].[C02_Details] a (NOLOCK)
WHERE
    (Sender_Account_No = @FromId) AND
    (Receive_Account_no = @ToId) AND
    (Transaction_Amount >= @AmountThreshold OR
@AmountThreshold IS NULL)) allChannels
ORDER BY
    CASE WHEN @SortField IS NULL THEN SerialNumber
END,
    CASE WHEN @SortField = 'SerialNumber' AND @SortOrder =
1 THEN SerialNumber END,
    CASE WHEN @SortField = 'TransactionDateTime' AND
@SortOrder = 1 THEN TransactionDateTime END,
    CASE WHEN @SortField = 'TransactionAmount' AND
@SortOrder = 1 THEN TransactionAmount END,
    CASE WHEN @SortField = 'GroupBy' AND @SortOrder = 1
THEN GroupBy END,
    CASE WHEN @SortField = 'ChannelId' AND @SortOrder = 1
THEN ChannelId END,
    CASE WHEN @SortField = 'AdditionalColumn1' AND
@SortOrder = 1 THEN AdditionalColumn1 END,
    CASE WHEN @SortField = 'AdditionalColumn2' AND
@SortOrder = 1 THEN AdditionalColumn2 END,
```

```

CASE WHEN @SortField = 'SerialNumber' AND @SortOrder =
0 THEN SerialNumber END DESC,
CASE WHEN @SortField = 'TransactionDateTime' AND
@SortOrder = 0 THEN TransactionDateTime END DESC,
CASE WHEN @SortField = 'TransactionAmount' AND
@SortOrder = 0 THEN TransactionAmount END DESC,
CASE WHEN @SortField = 'GroupBy' AND @SortOrder = 0
THEN GroupBy END DESC,
CASE WHEN @SortField = 'ChannelId' AND @SortOrder = 0
THEN ChannelId END DESC,
CASE WHEN @SortField = 'AdditionalColumn1' AND
@SortOrder = 0 THEN AdditionalColumn1 END DESC,
CASE WHEN @SortField = 'AdditionalColumn2' AND
@SortOrder = 0 THEN AdditionalColumn2 END DESC

OFFSET @Offset ROWS
FETCH NEXT @Fetch ROWS ONLY;

```

2. To Account From Account

a. Details

Details

Link Analysis Id: 2

From Property Id: Receiver Account No

To Property Id: Sender Account No

Graph Query: DECLARE @report TABLE (
Transaction_Serial_No BIGINT
Primary Key,
Transaction_Datetime DATETIME,
Transaction_Amount DECIMAL(19,2),
FromId NVARCHAR(25),
ToId NVARCHAR(25)
)

Table Query: SELECT * FROM (SELECT
Transaction_Serial_No as
SerialNumber,
Transaction_Datetime as
TransactionDateTime,
Transaction_Amount as
TransactionAmount,
FromId as GroupBy,
ChannelId as ChannelId,
AdditionalColumn1 as AdditionalColumn1,
AdditionalColumn2 as AdditionalColumn2
FROM Transaction)

Additional Column 1: ☒ Currency Code

Additional Column 2: ☒ Transaction Code

Save Delete Cancel

b. Graph Query

Graph Query

```

DECLARE @report TABLE (
    Transaction_Serial_No BIGINT Primary Key,
    Transaction_Datetime DATETIME,
    Transaction_Amount DECIMAL(19,2),
    FromId NVARCHAR(25),
    ToId NVARCHAR(25)
)

INSERT INTO @report (
    Transaction_Serial_No,
    Transaction_Datetime,
    Transaction_Amount,
    FromId,
    ToId
)

```

```

Transaction_Amount,
FromId,
ToId
)
SELECT
Transaction_Serial_No,
Transaction_Datetime,
Transaction_Amount,
Sender_Account_No,
Reciever_Account_No
FROM [dbo].[C01_Details] a (NOLOCK)
WHERE
(Reciever_Account_No = @Id) AND
(Transaction_Amount >= @AmountThreshold OR
@AmountThreshold IS NULL) AND
Transaction_Datetime >= @From AND
Transaction_Datetime <= @To
UNION
SELECT
Transaction_Serial_No,
Transaction_Datetime,
Transaction_Amount,
Sender_Account_No,
Receive_Account_no
FROM [dbo].[C02_Details] a (NOLOCK)
WHERE
(Receive_Account_no = @Id) AND
(Transaction_Amount >= @AmountThreshold OR
@AmountThreshold IS NULL) AND
Transaction_Datetime >= @From AND
Transaction_Datetime <= @To

SELECT
FromId,
ToId,
SUM(Transaction_Amount) AS [Total],
COUNT(*) as [Count]
FROM @report
GROUP BY FromId, ToId

```

c. Table Query

```

SELECT * FROM (SELECT
Transaction_Serial_No as SerialNumber,
Transaction_Datetime as TransactionDateTime,
Transaction_Amount as TransactionAmount,
Card_No as GroupBy,
'Source Account No' as GroupByColumnName,
'Internet Banking C01' as ChannelId,

```

```

        Currency_Code as AdditionalColumn1,
        Transaction_Code as AdditionalColumn2
    FROM [dbo].[C01_Details] a (NOLOCK)
    WHERE
        (Sender_Account_No = @FromId) AND
        (Reciever_Account_No = @ToId) AND
        (Transaction_Amount >= @AmountThreshold OR
@AmountThreshold IS NULL)
    UNION
    SELECT
        Transaction_Serial_No as SerialNumber,
        Transaction_Datetime as TransactionDateTime,
        Transaction_Amount as TransactionAmount,
        Account_No as GroupBy,
        'Source Account Number' as GroupByColumnName,
        'Cards C02' as ChannelId,
        Currency_Code as AdditionalColumn1,
        Transaction_Code as AdditionalColumn2
    FROM [dbo].[C02_Details] a (NOLOCK)
    WHERE
        (Sender_Account_No = @FromId) AND
        (Receive_Account_no = @ToId) AND
        (Transaction_Amount >= @AmountThreshold OR
@AmountThreshold IS NULL)) allChannels
    ORDER BY
        CASE WHEN @SortField IS NULL THEN SerialNumber
END,
        CASE WHEN @SortField = 'SerialNumber' AND @SortOrder =
1 THEN SerialNumber END,
        CASE WHEN @SortField = 'TransactionDateTime' AND
@SortOrder = 1 THEN TransactionDateTime END,
        CASE WHEN @SortField = 'TransactionAmount' AND
@SortOrder = 1 THEN TransactionAmount END,
        CASE WHEN @SortField = 'GroupBy' AND @SortOrder = 1
THEN GroupBy END,
        CASE WHEN @SortField = 'ChannelId' AND @SortOrder = 1
THEN ChannelId END,
        CASE WHEN @SortField = 'AdditionalColumn1' AND
@SortOrder = 1 THEN AdditionalColumn1 END,
        CASE WHEN @SortField = 'AdditionalColumn2' AND
@SortOrder = 1 THEN AdditionalColumn2 END,
        CASE WHEN @SortField = 'SerialNumber' AND @SortOrder =
0 THEN SerialNumber END DESC,
        CASE WHEN @SortField = 'TransactionDateTime' AND
@SortOrder = 0 THEN TransactionDateTime END DESC,
        CASE WHEN @SortField = 'TransactionAmount' AND
@SortOrder = 0 THEN TransactionAmount END DESC,
        CASE WHEN @SortField = 'GroupBy' AND @SortOrder = 0
THEN GroupBy END DESC,
        CASE WHEN @SortField = 'ChannelId' AND @SortOrder = 0

```

```

THEN ChannelId END DESC,
      CASE WHEN @SortField = 'AdditionalColumn1' AND
@SortOrder = 0 THEN AdditionalColumn1 END DESC,
      CASE WHEN @SortField = 'AdditionalColumn2' AND
@SortOrder = 0 THEN AdditionalColumn2 END DESC

OFFSET @Offset ROWS
FETCH NEXT @Fetch ROWS ONLY;

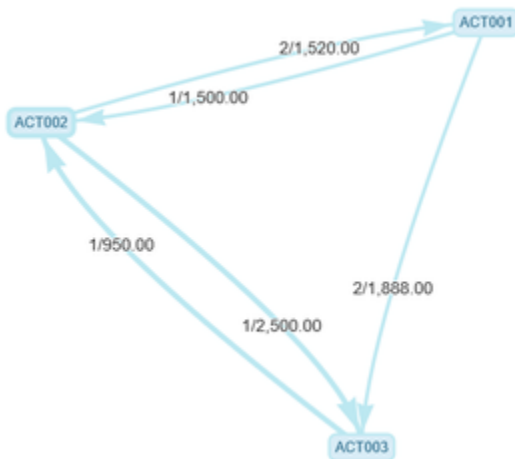
```

Review Link Analysis

1. go to Transaction > Alert Review
2. Select the Link Analysis Hyperlink
3. Click on Analyse



4. Click on the external node to view external linkage. (e.g ACT002)



5. Click on the connectors to view the transaction that is from and to the account (e.g 2/1888)

Transaction(s)

Sender Account No From

ACT001

Sender Account No To

ACT003

Serial No	Group By	Date & Time	Amount	Channel	Currency Code	Transaction Code
2	164421091120112	10/08/2018 12:00:00 PM	1,000.00	Internet Banking C01	AUD	101
5	164421091120112	10/08/2018 12:00:00 PM	888.00	Cards C02	AUD	101

Showing 1 to 2 of 2 records.

1

```

graph LR
    ACT001 -- "2/1,888.00" --> ACT003
    ACT003 -- "1/2,500.00" --> ACT002
    ACT002 -- "1/950.00" --> ACT003
    ACT002 -- "1/1,500.00" --> ACT001
    ACT002 -- "1/950.00" --> ACT003

```


Related articles

- [How to Setup Link Analysis](#)