Posting Query for US Process code

A screenshot of a computer program

Description automatically generated

The US process posting query involves below accounts:

10100,30100,30500,74102,74280,11101,82101

**Process codes:**

[Process].[Process Code].&[152] = US

[Process].[Process Code].&[135] = UA

[Process].[Process Code].&[136] = UB

**1. Account 10100 (Base Allocation, Data Stage 2)**

A screenshot of a computer

Description automatically generated

The scope focuses on Account 10100 in the Base Allocation (Data Stage 2) for SYND Platform entities. It operates by applying either:

Live Agresso (Data Stage 4) with a multiplier of -1, or

A combination of Gross Premium Top-Up and External Brokerage Percentage, depending on the conditions.

**Explanation:**

If the Premium Forecast (Gross Gross Premium) is 0 or empty, it retrieves Live Agresso data and multiplies it by -1.

Otherwise, it uses the Gross Premium Top-Up from Data Stage 2 (Account 10100) and multiplies it by the External Brokerage Percentage.

This =iif(

([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember) = 0

OR

ISEMPTY(([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember)),

([Account].[Account].&[30100],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].&[135], [Measures].CurrentMember ) \* -1,

(

([Account].[Account].&[10100],[Data Stage].[Data Stage].&[2],

[Process].[Process Code].&[152], [Measures].CurrentMember)

) \*

(

Divide(

([Account].[Account].&[2000022],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember),

([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember),

0

)

)

)

**2. Account 30100 (Base Allocation, Data Stage 2)**

A screenshot of a computer

Description automatically generated

This is a similar calculation for Account 30100 in Base Allocation (Data Stage 2) with focus on SYND Platform entities. It evaluates and performs the same Gross Premium Top-Up and External Brokerage Percentage logic.

**Explanation:**

It follows the same logic as Account 10100 with the fallback to Live Agresso (Data Stage 4) if the forecast premium is 0 or empty.

The Top-Up calculation and external brokerage percentage are used if the forecast is available.

This =iif(

([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember) = 0

OR

ISEMPTY(([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember)),

([Account].[Account].&[30100],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].&[135], [Measures].CurrentMember ) \* -1,

(

([Account].[Account].&[10100],[Data Stage].[Data Stage].&[2],

[Process].[Process Code].&[152], [Measures].CurrentMember)

) \*

(

Divide(

([Account].[Account].&[2000022],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember),

([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember),

0

)

)

)

**3. Account 30500 (Base Allocation, Data Stage 2)**

This calculation handles Internal Commission for Account 30500 under Base Allocation (Data Stage 2). It is designed to account for internal commissions based on Gross Premium Top-Up and Internal Commission Percentage.

**Explanation:**

If the Premium Forecast (Gross Gross Premium) is 0 or empty, it takes the Live Agresso value and multiplies it by -1.

Otherwise, it calculates the value based on Gross Premium Top-Up and multiplies it by the Internal Commission Percentage.

This =

iif(

([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember) = 0

OR

ISEMPTY(([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember)),

([Account].[Account].&[30500],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].&[135], [Measures].CurrentMember ) \* -1,

(

([Account].[Account].&[10100],[Data Stage].[Data Stage].&[2],

[Process].[Process Code].&[152], [Measures].CurrentMember)

) \*

(

Divide(

([Account].[Account].&[2000023],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember),

([Account].[Account].&[2000021],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember),

0

)

)

)

**4. Account 74102 (Base Allocation, Data Stage 2)**

A screenshot of a computer program

Description automatically generated

The calculation for Account 74102 focuses on combining the values of Accounts 10100, 30100, and 30500 in Base Allocation (Data Stage 2) for SYND Platform entities.

**Explanation:**

The values from Accounts 10100, 30100, and 30500 are summed and then multiplied by -1.

This =

(

(

([Account].[Account].&[10100],[Data Stage].[Data Stage].&[2],

[Process].[Process Code].&[152], [Measures].CurrentMember)

+

([Account].[Account].&[30100],[Data Stage].[Data Stage].&[2],

[Process].[Process Code].&[152], [Measures].CurrentMember)

+

([Account].[Account].&[30500],[Data Stage].[Data Stage].&[2],

[Process].[Process Code].&[152], [Measures].CurrentMember)

) \* -1

)

**5. Account 74280 (Base Allocation, Data Stage 2)**

This calculation focuses on Account 74280, following similar logic to Account 74102 but likely applies a different adjustment or process code combination, depending on the specifics of the scenario.

**Explanation:**

It aggregates the same accounts as 74102, applying further adjustments or different multipliers.

The exact difference from 74102 would depend on the specific calculation context and platform configuration.

**6. Account 11101 (Base Allocation, Data Stage 2)**

A screenshot of a computer code

Description automatically generated

The calculation for Account 11101 applies specific deductions from Live Agresso (Data Stage 4) and references values from Account 2000024 in External Data (Data Stage 1).

**Explanation:**

It starts with the value from Account 2000024 and subtracts allocations from Live Agresso (Data Stage 4) under specific Process Codes UB, 2D, and 2H.

This =

([Account].[Account].&[2000024],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember)

-

([Account].[Account].&[11101],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].&[136], [Measures].CurrentMember)

-

([Account].[Account].&[11101],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].[2D], [Measures].CurrentMember )

-

([Account].[Account].&[11101],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].[2H], [Measures].CurrentMember)

**7. Account 82101 (Base Allocation, Data Stage 2)**

A screenshot of a computer program

Description automatically generated

This is the inverse calculation of Account 11101. It calculates similar deductions but multiplies the entire result by -1.

**Explanation:**

Like Account 11101, it subtracts values from Live Agresso (Data Stage 4) under specific Process Codes but applies the result to Account 82101 and multiplies by -1.

This =

(

([Account].[Account].&[2000024],[Data Stage].[Data Stage].&[1],

[Process].[Process Code].&[152], [Measures].CurrentMember)

-

([Account].[Account].&[11101],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].&[136], [Measures].CurrentMember)

-

([Account].[Account].&[11101],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].[2D], [Measures].CurrentMember )

-

([Account].[Account].&[11101],[Data Stage].[Data Stage].&[4],

[Process].[Process Code].[2H], [Measures].CurrentMember )

) \* -1