



Universidad
Carlos III de Madrid

TECNOLOGÍAS *DEL* SECTOR FINANCIERO

Práctica 11: CEP

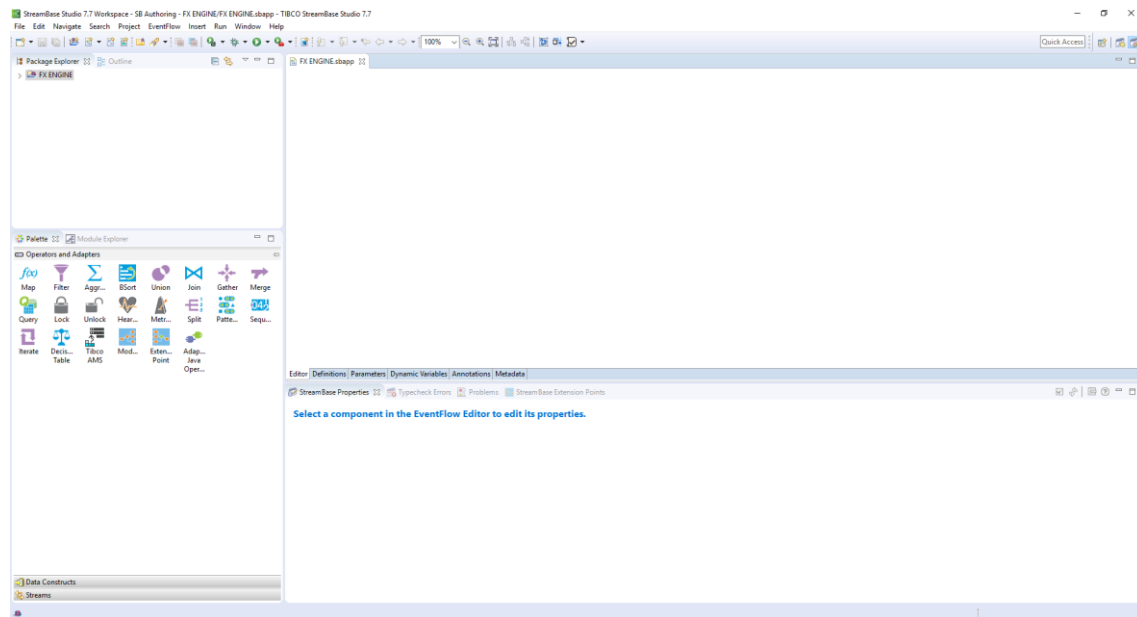
Andoni Alcelay

MÁSTER UNIVERSITARIO EN TECNOLOGÍAS DEL SECTOR FINANCIERO: FINTECH

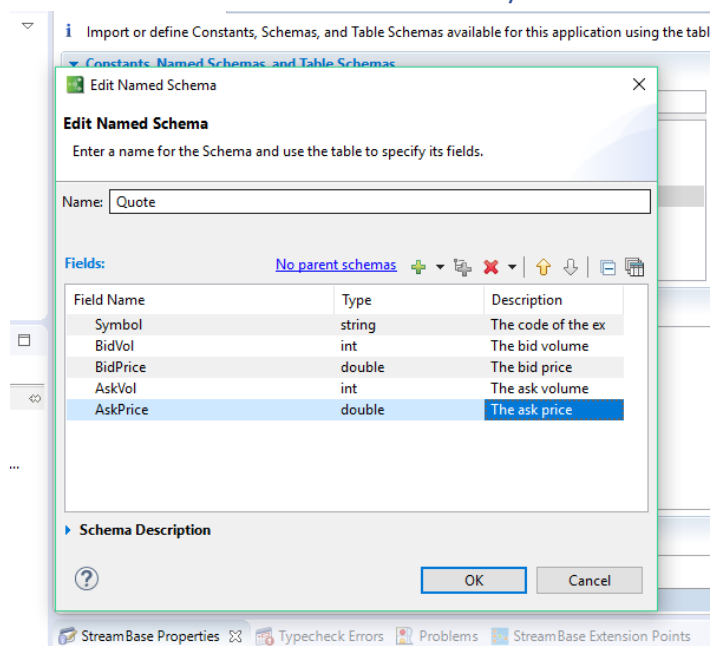
Introducción

En este documento se mostrará el procedimiento que se ha seguido para realizar un proyecto con Streambase Studio y el resultado de la ejecución de este:

Montaje del proyecto



Creación de variables Quote y Order









Constants, Named Schemas, and Table Schemas

Edit Named Schema


Enter a name for the Schema and use the table to specify its fields.

Name:

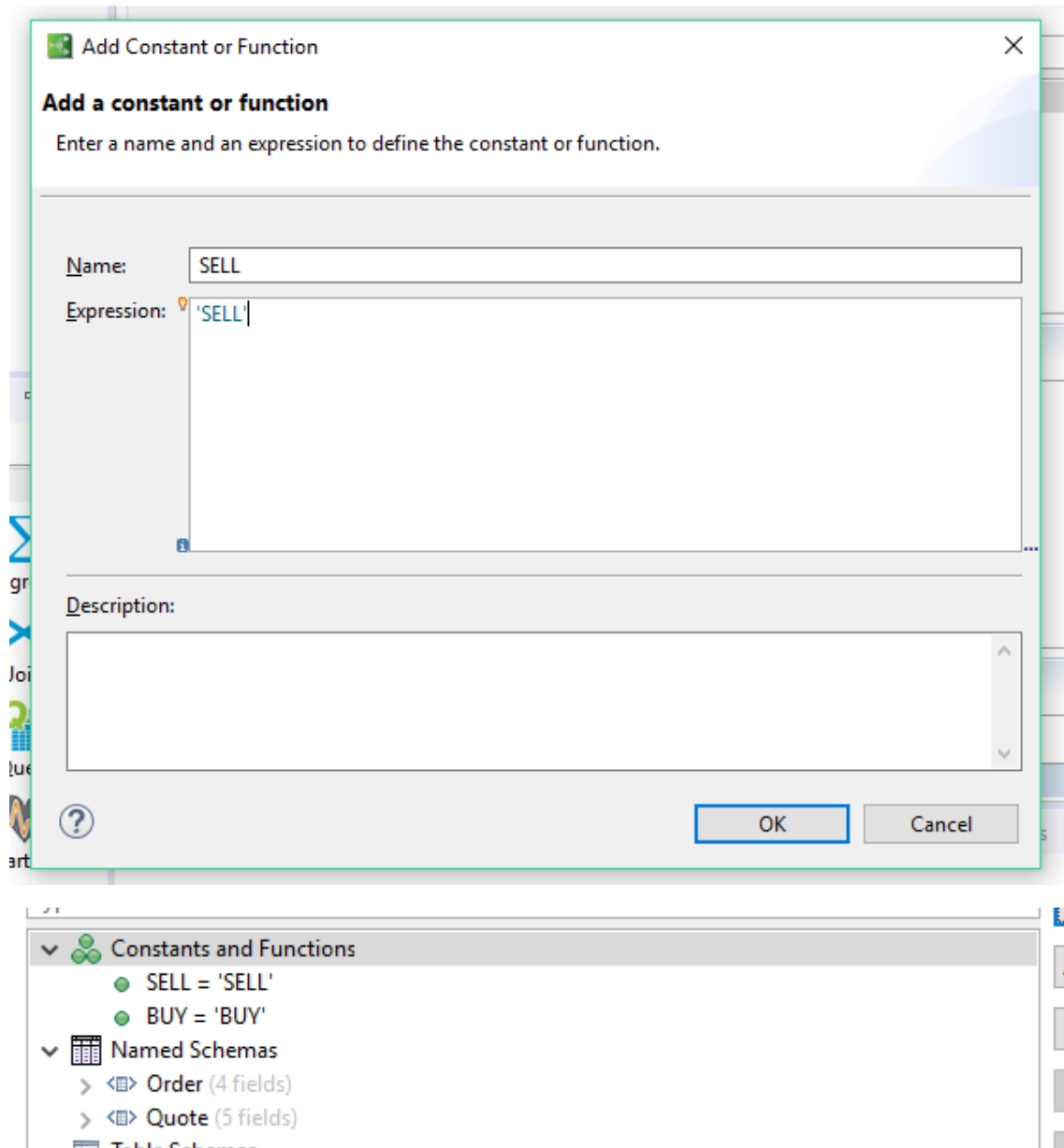
Fields: [No parent schemas](#)      

Field Name	Type	Description
Symbol	string	The code
Vol	int	Volume
Price	double	The price of the or...
Side	string	Determines if is a b...

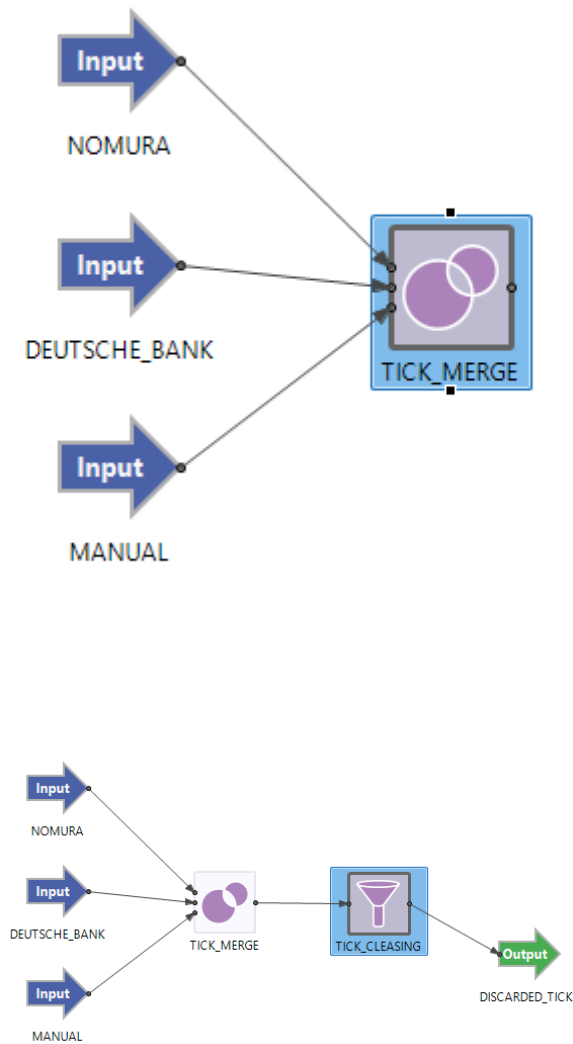
Schema Description



Creación de constantes



Creación de filtros y outputs



DefinitionsParametersDynamic VariablesAnnotationsMetadata

K_CLEASING - StreamBase PropertiesTypecheck ErrorsProblemsStreamBase Extension Points

Specify at least one predicate by adding and completing rows in the table below.

il

Predicate SettingsConcurency

ate output port for non-matching tuples

cates:

ut Port

Predicate

(Symbol == 'EURUSD' || Symbol == 'EURCHF')&&(BidPrice > 0 && AskPrice > 0)...

StreamsFunctionsExpression QuickRef

InputOutput

input1 (5 fields)

Symbol string [The code of the ex]

BidVol int [The bid volume]

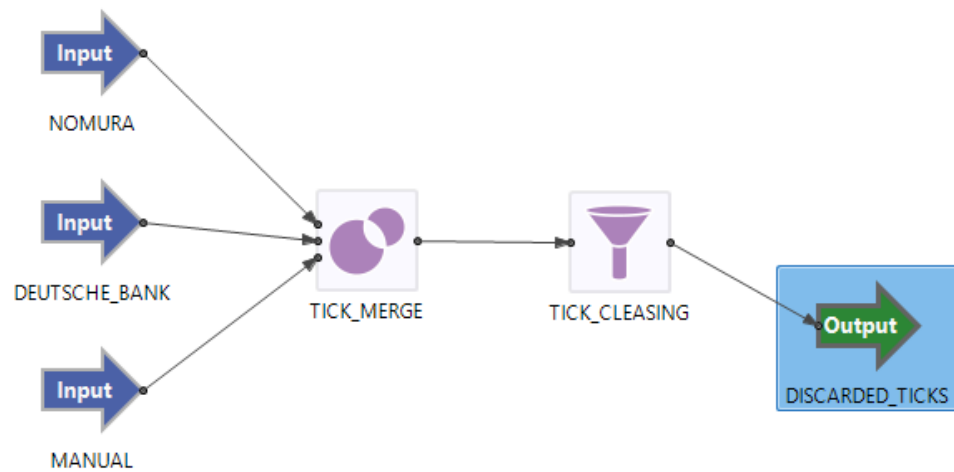
BidPrice double [The bid price]

AskVol int [The ask volume]

AskPrice double [The ask price]

Constants (2)

3



Definitions Parameters Dynamic Variables Annotations Metadata

DISCARDED_TICKS - StreamBase Properties Typecheck Errors Problems StreamBase Extension Points

al Schema Advanced

out Schema: ☒ Let typechecking define ☐ Declare:

ent Schema:

Field Name	Type
Symbol	string
BidVol	int
BidPrice	double
AskVol	int
AskPrice	double

Streams Functions

Input Output

input1 (5 fields)

- Symbol string [The code of the symbol]
- BidVol int [The bid volume]
- BidPrice double [The bid price]
- AskVol int [The ask volume]
- AskPrice double [The ask price]

Constants (2)

Creación de los where y las tablas de BID y ASK

The screenshot displays the StreamBase IDE interface. The top part shows a data flow diagram with the following components and connections:

- Inputs:** Three blue arrow components labeled "NOMURA", "DEUTSCHE_BANK", and "MANUAL".
- TICK_MERGE:** A purple circle component receiving input from the three input components.
- TICK_CLEANSING:** A purple funnel component receiving input from TICK_MERGE.
- Split:** A grey component with three outputs (1, 2, 3) receiving input from TICK_CLEANSING.
- Query:** A blue component with a green 'Q' icon receiving input from the Split component.
- Outputs:** A green arrow component labeled "DISCARDED_TICKS" receiving input from TICK_CLEANSING. Two database table icons labeled "BID_LIQUIDITY" and "ASK_LIQUIDITY" are connected to the Query component via dashed lines.

The bottom part of the screenshot shows the "Query - StreamBase Properties" panel with the following configuration:

General | **Query** | **Operation (Insert)** | **Fallback** | **Output** | **Group Options**

Associated Table: BID_LIQUIDITY
Operation: * Write
Where: * Primary Index (BidPrice, Syn

***Matches:**

Index Expressi...	Type	Match Expression
BidPrice	double	input.Symbol
Symbol	string	input.BidPrice

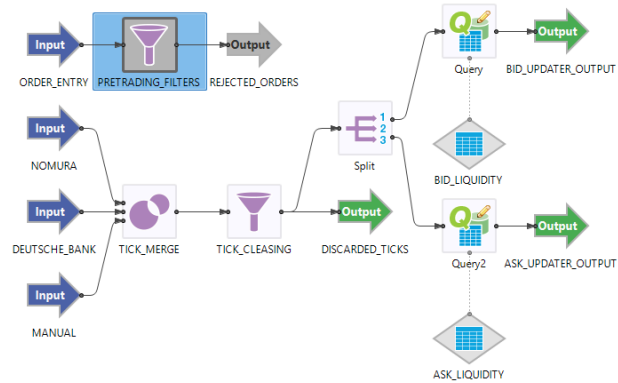
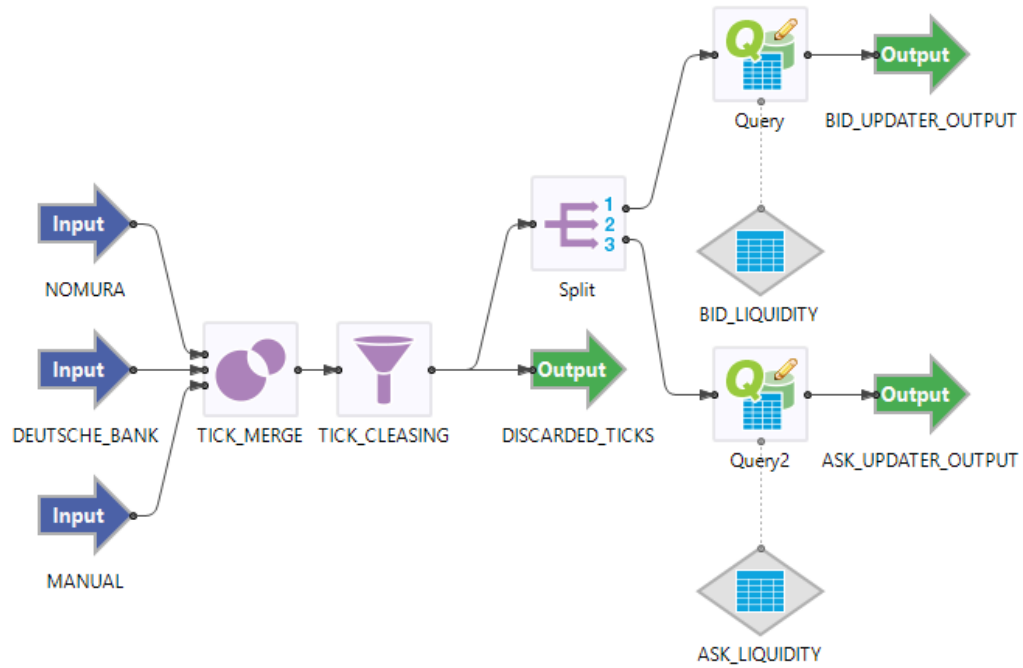
Streams | **Functions** | **Expression QuickRef**

Input | **Output**

Associated Table (5 fields)

- Symbol string [The code of the ex]
- BidVol int [The bid volume]
- BidPrice double [The bid price]
- AskVol int [The ask volume]
- AskPrice double [The ask price]

Constants (2)



ditor | Definitions | Parameters | Dynamic Variables | Annotations | Metadata

PRETRADING_FILTERS - StreamBase Properties | Typecheck Errors | Problems | StreamBase Extension Points

Specify at least one predicate by adding and completing rows in the table below.

General | Predicate Settings | Concurrency

Create output port for non-matching tuples

Predicates:

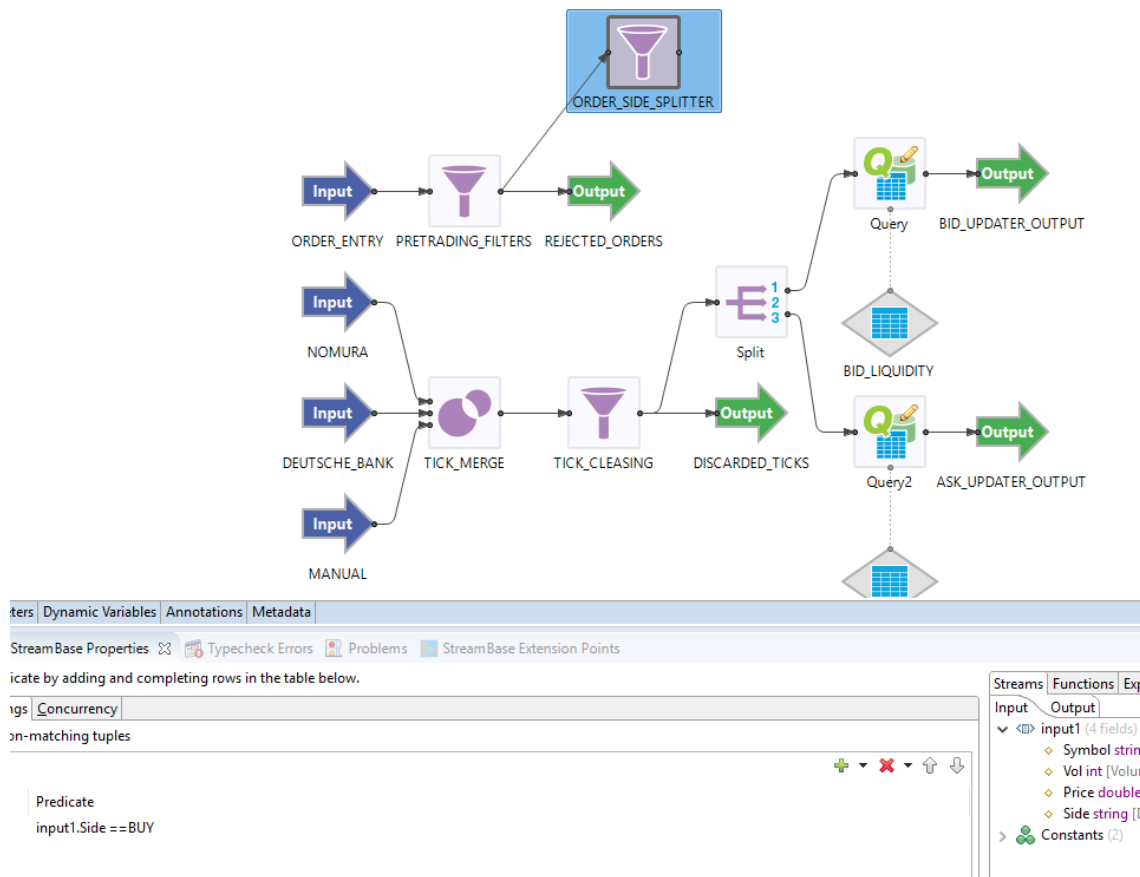
Output Port	Predicate
1	<pre> (Symbol == 'EURUED' Symbol == 'EURCHF') && (Vol >= 30 && Vol <= 150) && (Side == BUY Side == SELL) </pre>

Streams | Functions | Expression QuickRef

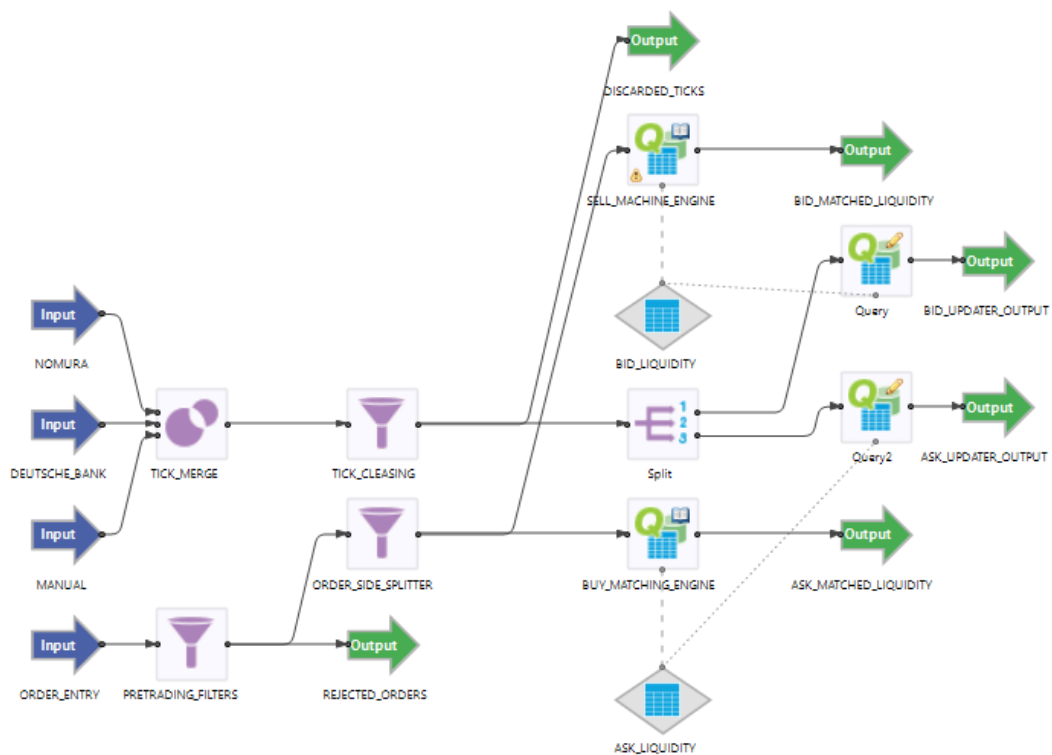
Input | Output

input1 (4 fields)

- Symbol string [The code]
- Vol int [Volume]
- Price double [The price of the order]
- Side string [Determines if is a buy or sell]
- Constants (2)



Matching de órdenes contra el Liquidity Book



Ejecución del programa

[illegible]

Application Output Application Input Profiler Variables

Stream: (All Output Streams) select...

Time	Output Stream	Fields
14:14:47	DISCARDED_TICKS	Symbol=EURUSD, BidVol=1000, BidPrice=0.75, AskVol=2000, AskPrice=0.85
14:14:47	ASK_UPDATER_OUTPUT	Symbol=EURUSD, BidVol=1000, BidPrice=0.75, AskVol=2000, AskPrice=0.85
14:14:47	BID_UPDATER_OUTPUT	Symbol=EURUSD, BidVol=1000, BidPrice=0.75, AskVol=2000, AskPrice=0.85