Introduction:
A testing tool. It will be capable of

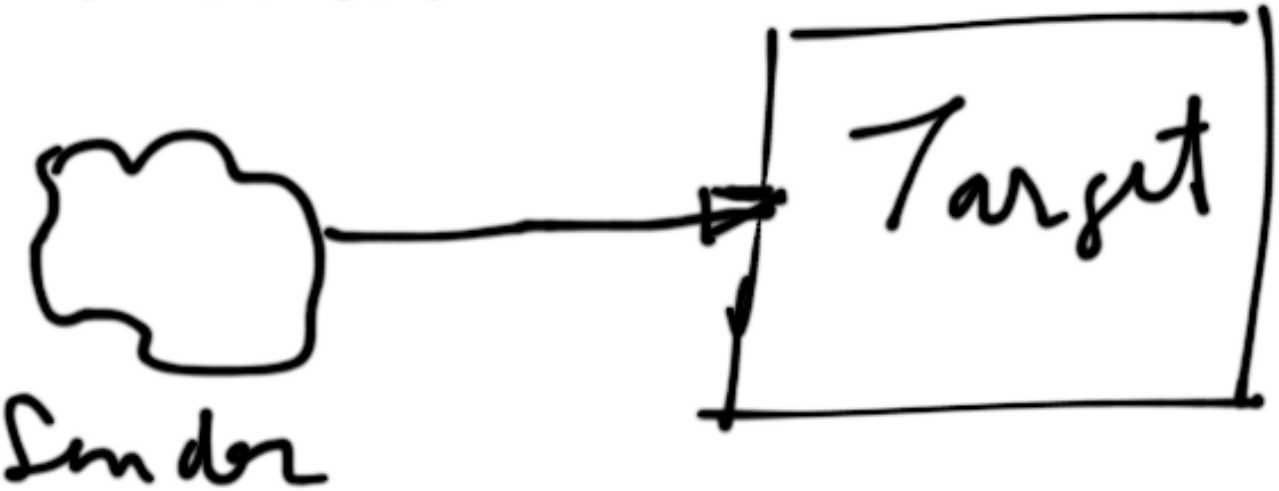
- sending / receiving financial messages
- user can add new message types of data which can be sent or received by the tool.
- Will be able to create & store testcases based on them messages.
- Will be able to compare responses with baselines
- Will provide requested reports of the test execution
- User can configure ie what network protocol it wants to use for sending / receiving

To achieve the stated goals for a testing tool we define a testing entity. This entity will be used to send or receive the data as per requirement.

There can be three types of such an entity

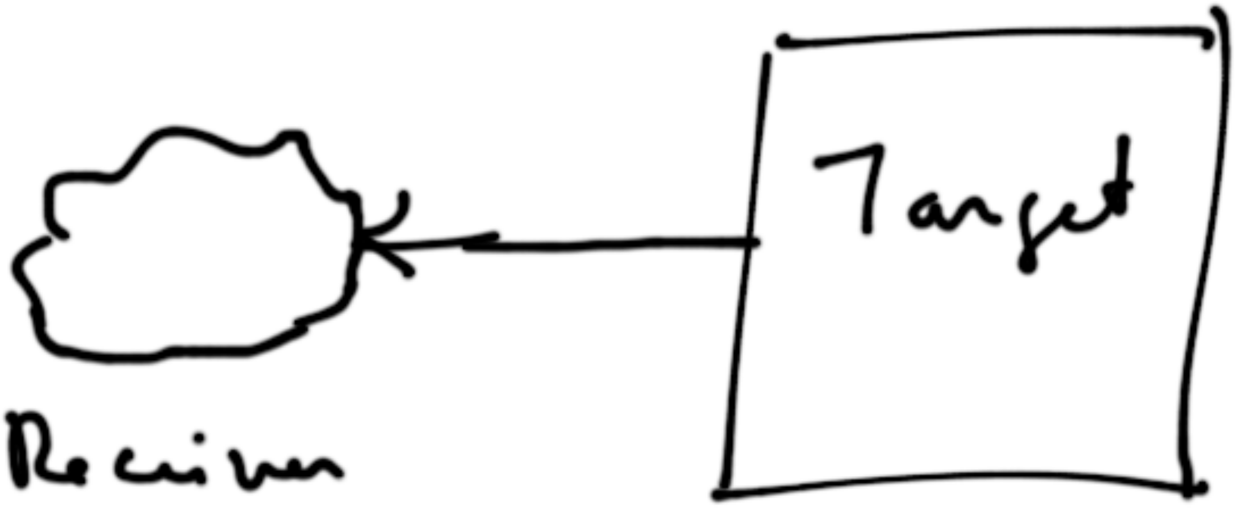
1. sender
2. receiver
3. both

Sender:



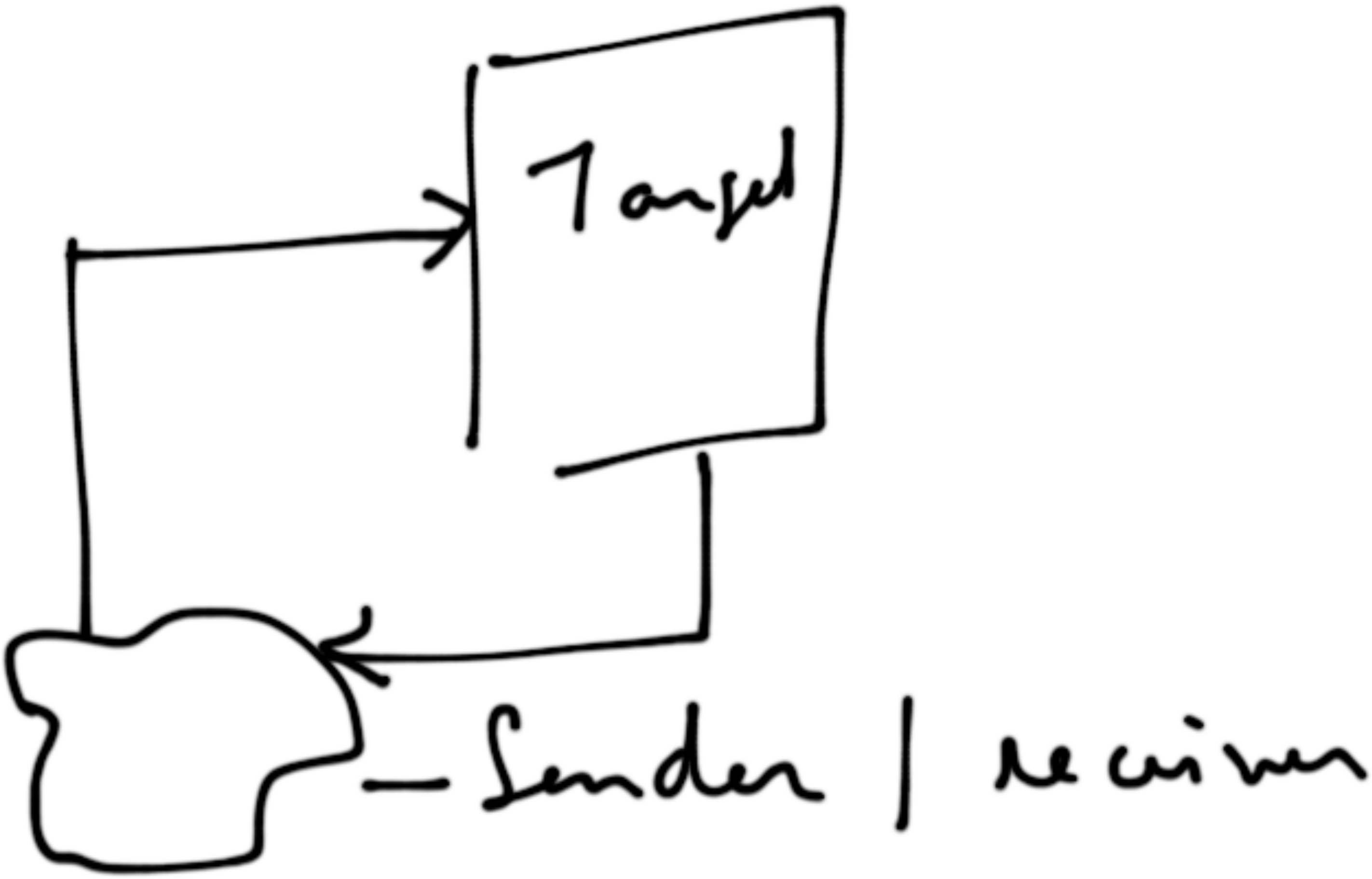
- Capable of sending msgs only
- Can send any configured msg
- Supports tcp, udp, mcast, ftp etc.
- Supports multiple clients

Receiver :



- Can only receive msgs
- Can receive configured msgs
    - Can associate event with receipt of any message
    - The event will be executed as soon as th message is received
- It can be used to record & store msgs.

Both:



- Contains sender & receiver characteristics

Configuration of an entity:

Common-

1. Name
2. Msgs Allowed
3. Network communication
4. Label to which this entity belongs

for eg :

Market data entity :

Name - market data

Type - sender

Msgs supported - cqs, cts, opra,
nqds

Network - protocol - mcast
tcp
udp

hostname -

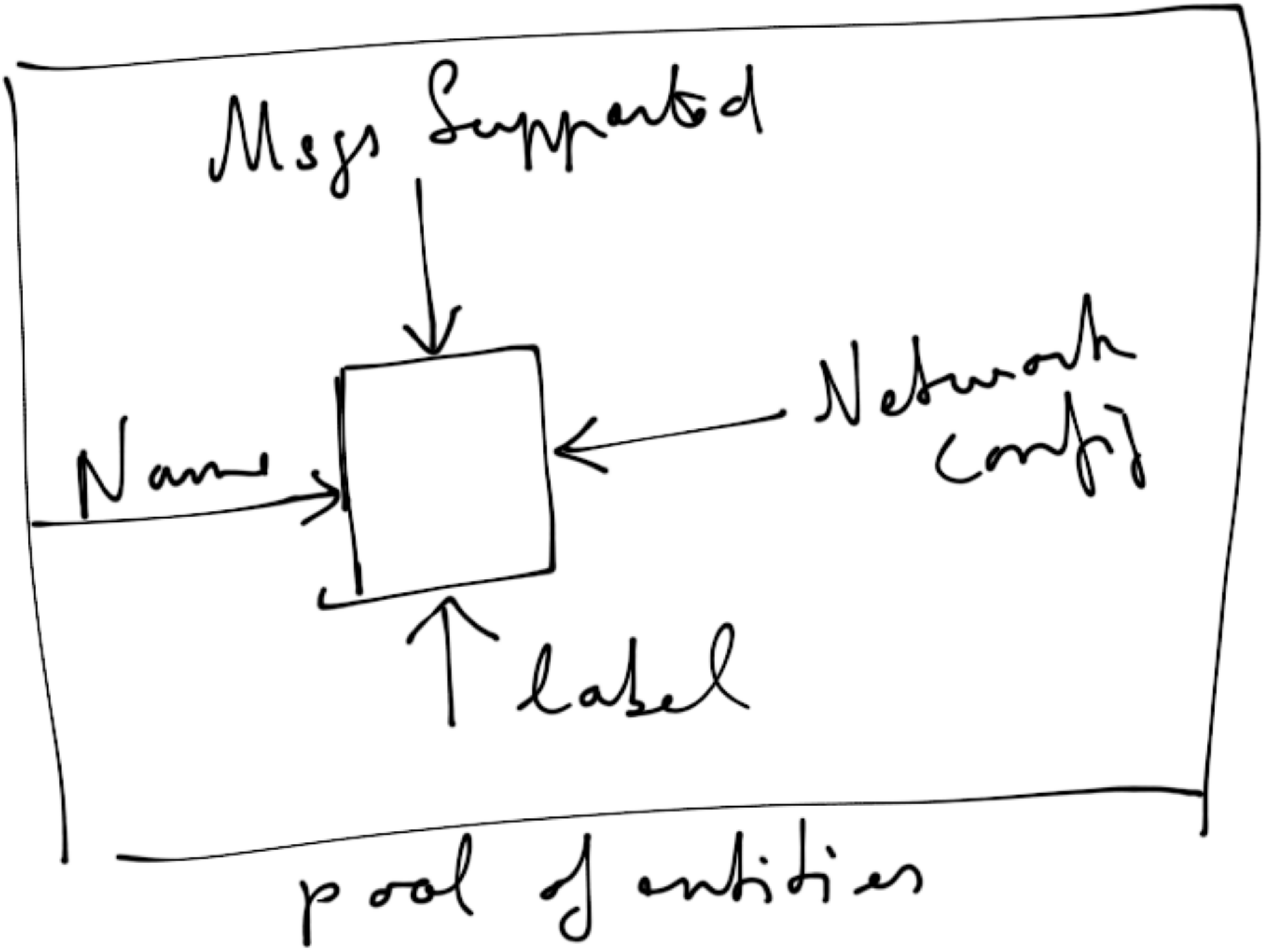
port -

Label -

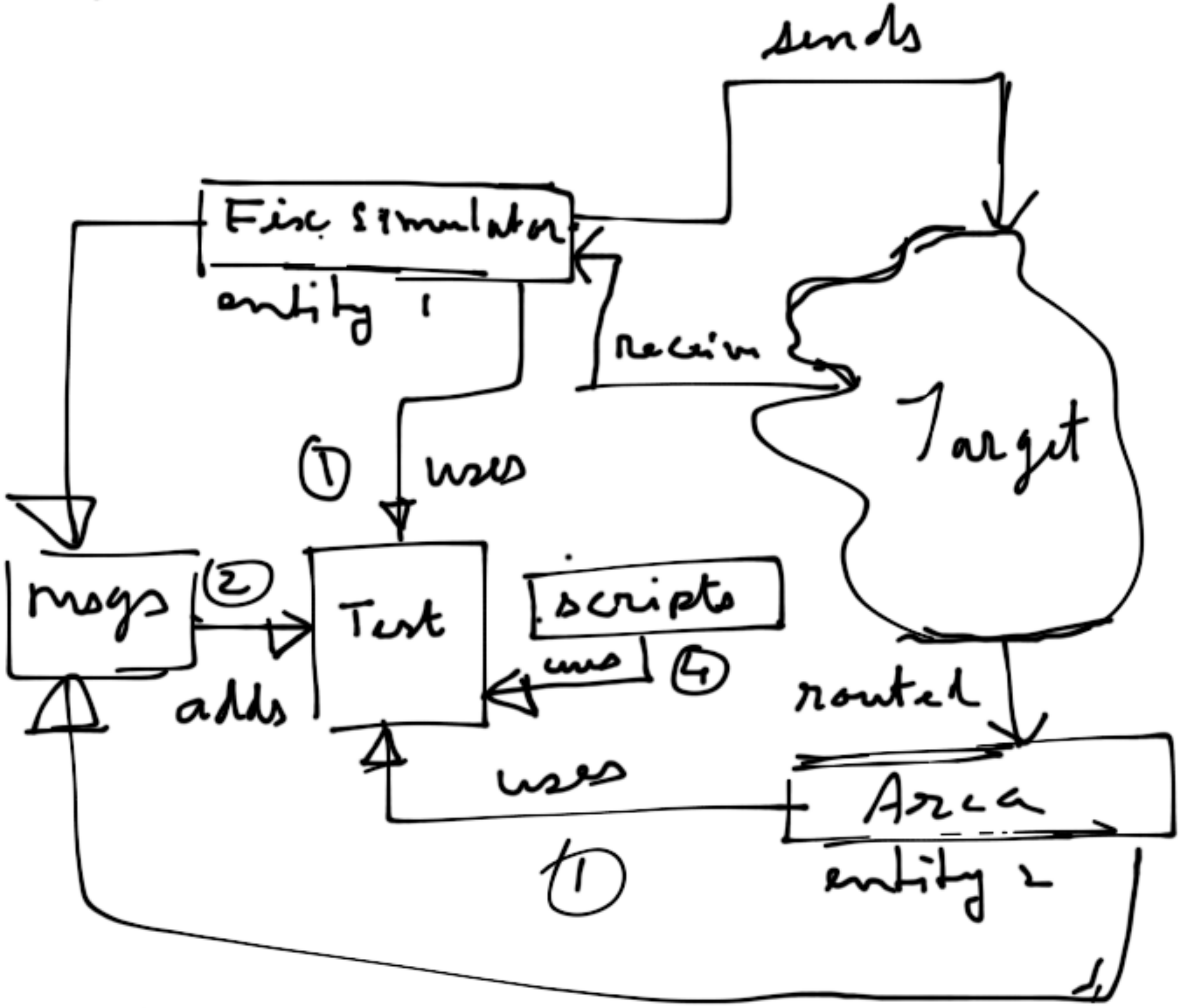
Authorized for - qa

Typical Work flow :

1. Admin creates the entity & stores it to a pool of entities



2. Tester.
Will avail of the entities he wants to use.



① Tester adds Fix Simulator & Arca entities to his test case

② Adds messages from the

msg pools supported by the two entities.

③ Adds scripts to be executed between / after the messages as per test case requirements.

④ Sets up expected results

⑤ Runs the test case

⑥ Compares expected vs actual

⑦ Stores the project with the test case.