

io-topics publish: quickly deploy publishers on demand

If you need to quickly deploy a bunch services for line-based or fixed-width data over TCP, local sockets, ZeroMQ, etc, now you can use **io-topics**, a utility in **comma**. You can deploy services that run continuously or start only in case if there is at least one client (e.g. if they are too resource greedy).

Perhaps, it is not a replacement for a more proper middleware like ROS or simply systemd, but the advantages of **io-publish-topics** are its light weight, ad-hoc nature, ability to run a mix of transport protocols.

Try the following toy example of **io-topics publish**:

```
> # run publisher with topics a and b, with b on demand
> io-topics publish --config <( echo "a/command=csv-paste line-number"; echo "a/port=8888"; echo "b/command=csv-
paste line-number"; echo "b/port=9999"; echo "b/on_demand=1" )
io-topics: publish: will run 'comma_execute_and_wait --group' with commands:
io-topics: publish:   io-publish tcp:8888   -- csv-paste line-number
io-topics: publish:   io-publish tcp:9999   --on-demand -- csv-paste line-number

> # in a different shell, observe that topic a keeps running even if no-one is listening,
> # whereas topic b runs only if at least one client is connected:
> socat tcp:localhost:8888 | head -n5 # will output something like, since the service keeps running even if
there are no clients connected:
16648534
16648535
16648536
16648537
16648538

> socat tcp:localhost:9999 - | head -n5 # whenever the first client connects, will start from 0, since it runs
only if at least one client is connected
0
1
2
3
4
```

You also can create - on the fly, if you want - a light-weight subscriber, as in example below. Run publishing as in the example above and then run **io-topics cat**:

```
> io-topics cat --config <( echo "a/command=head -n5 > a.csv"; echo "a/address=tcp:localhost:8888"; echo "b
/command=head -n5 > b.csv"; echo "b/address=tcp:localhost:9999" )
io-topics: cat: will run 'comma_execute_and_wait --group' with commands:
io-topics: cat:   bash -c io-cat tcp:localhost:8888 | head -n5 > a.csv
io-topics: cat:   bash -c io-cat tcp:localhost:9999 | head -n5 > b.csv
> # check output
> cat a.csv
203740462
203740463
203740464
203740465
203740466
> cat b.csv
0
1
2
3
4
```