

Q&A
is the answer

Ian Barber

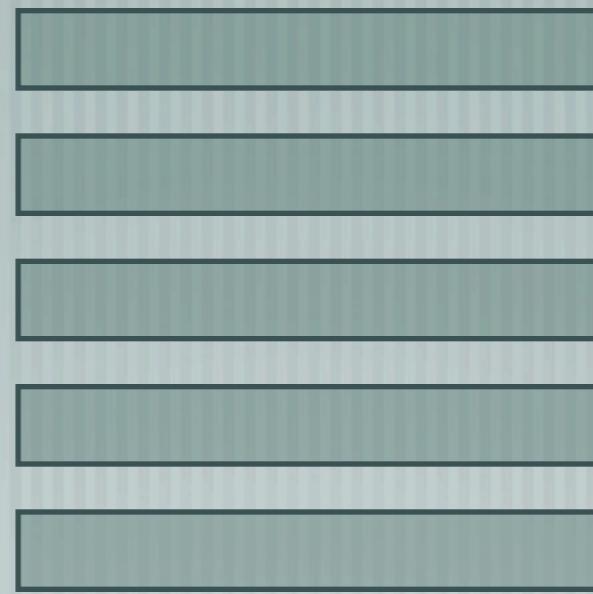
<http://joind.in/talk/view/2523>

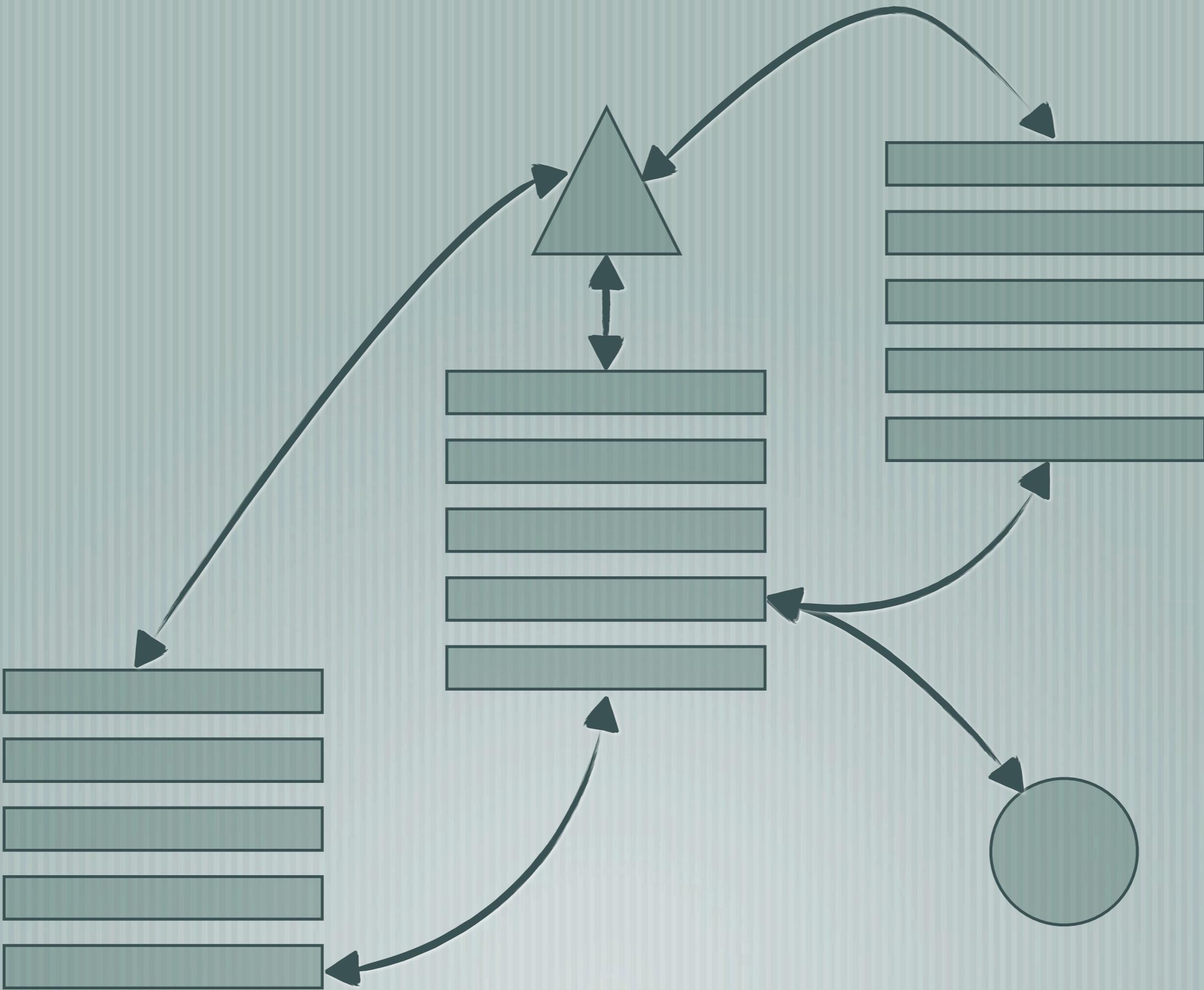
<http://phpir.com>

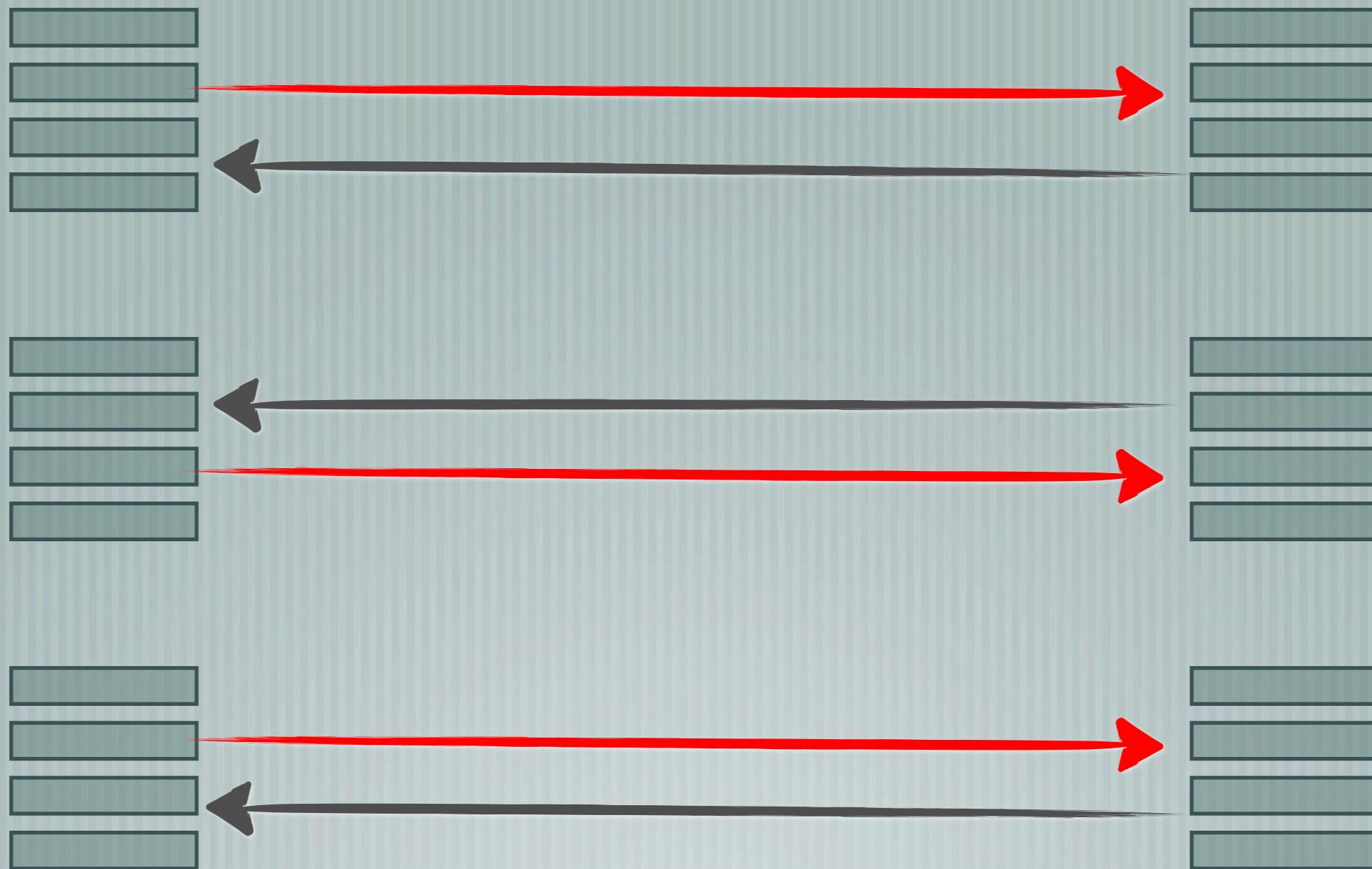
ian.barber@gmail.com @ianbarber

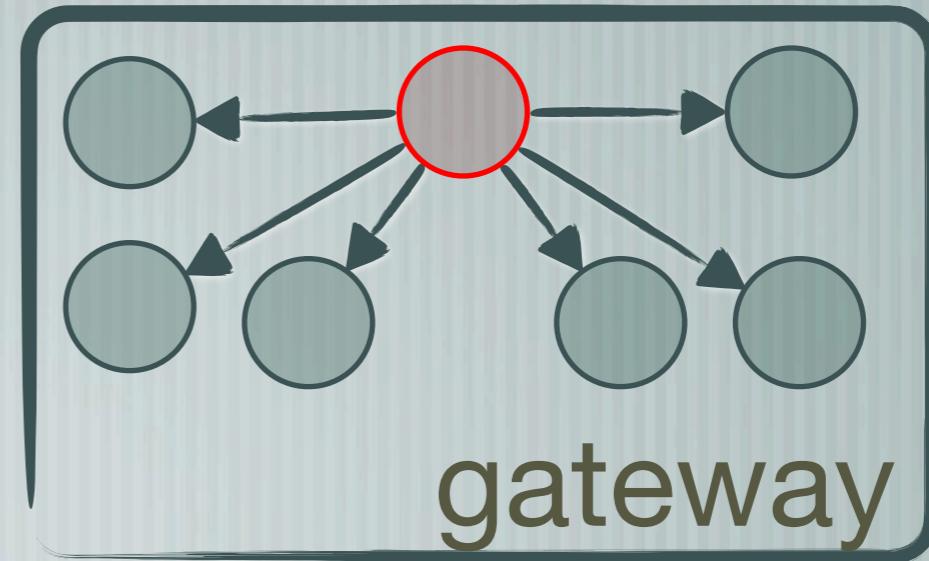
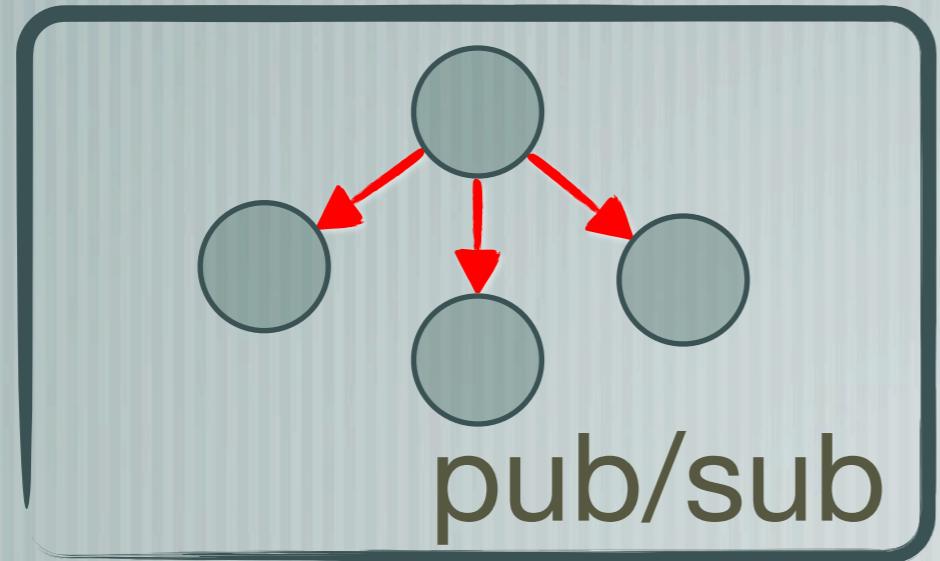
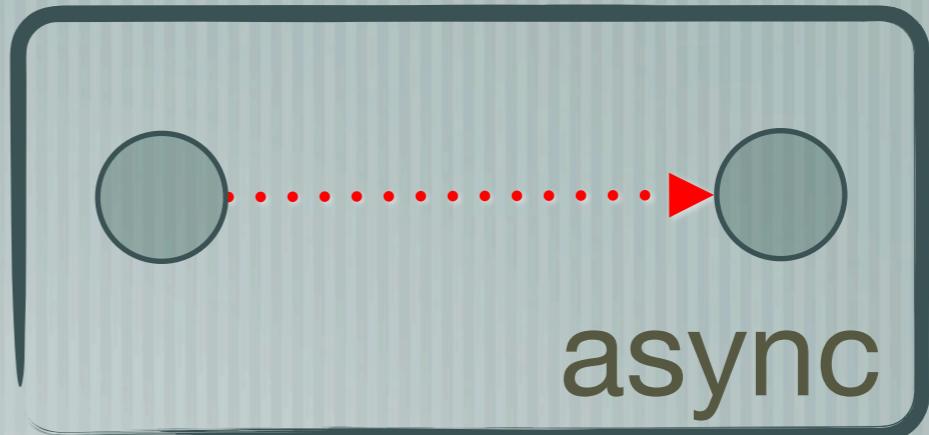
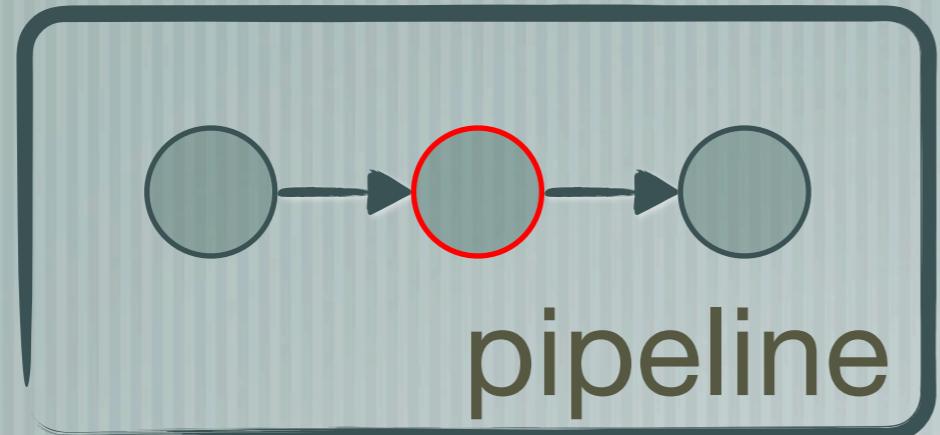
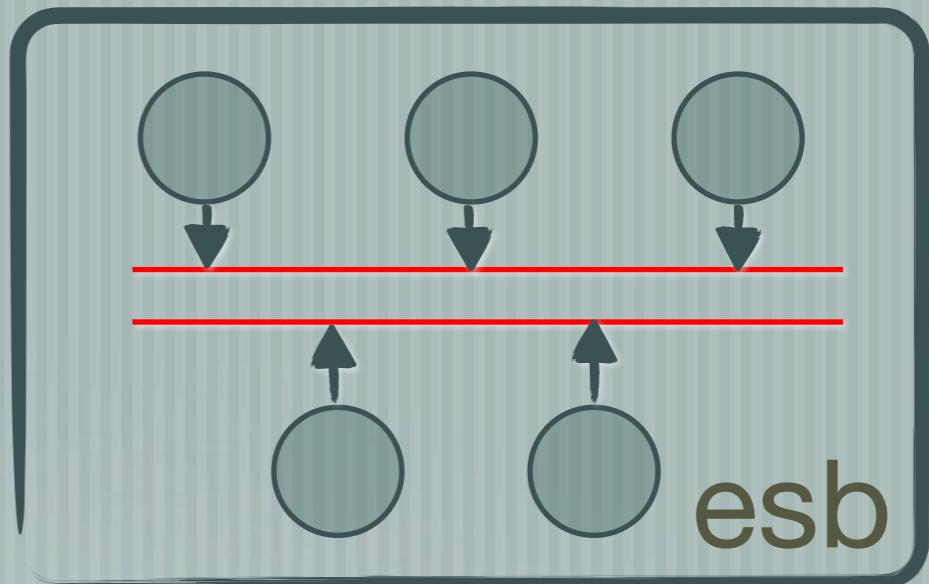
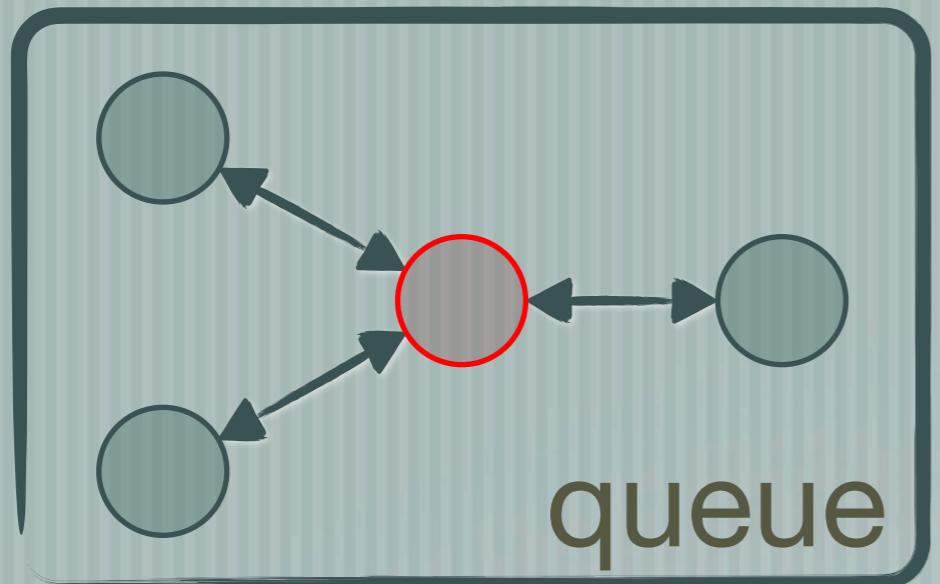
**“OMQ is
unbelievably cool
– if you haven’t
got a project that
needs it, make
one up”**

jon gifford - loggly









request/response



```
$ctx = new ZMQContext();
$server =
    new ZMQSocket($ctx, ZMQ::SOCKET REP);
$server->bind("tcp://*:5454");

while(true) {
    $message = $server->recv();
    $server->send($message . " World");
}
```

rep.php

request/response



```
$ctx = new ZMQContext();
$req =
    new ZMQSocket($ctx, ZMQ::SOCKET_REQ);
$req->connect("tcp://localhost:5454");

$req->send("Hello");
echo $req->recv();
```

req.php

rep.py

```
import zmq

context = zmq.Context()
server = context.socket(zmq.REP)
server.connect("tcp://localhost:5455")

while True:
    message = server.recv()
    print "Sending", message, "World\n"
    server.send(message + " World")
```

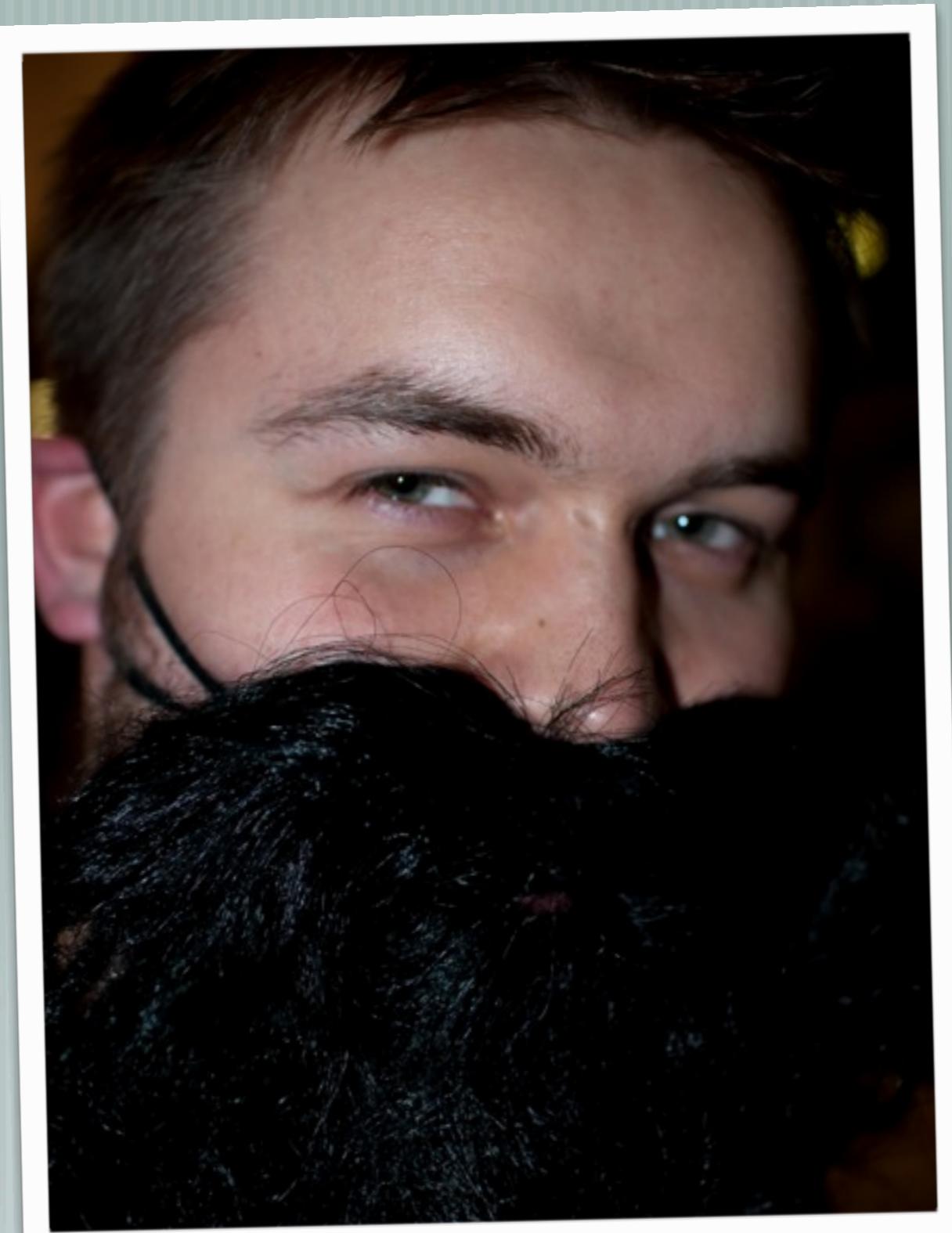


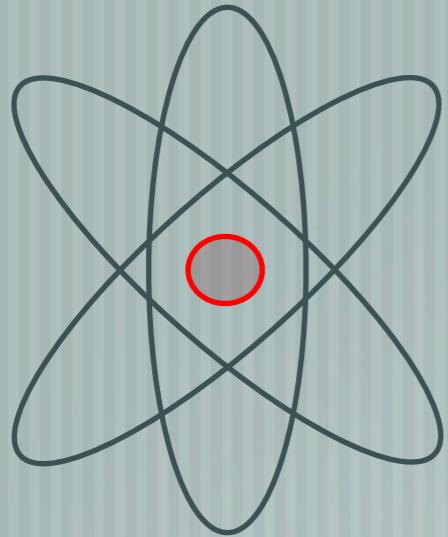
Image: http://flickr.com/photos/sebastian_bergmann/3318754086

```
wget http://download.zeromq.org/  
zeromq-2.1.1.tar.gz  
tar xvzf zeromq-2.1.1.tar.gz  
cd zeromq-2.1.1/  
.configure  
make  
sudo make install
```

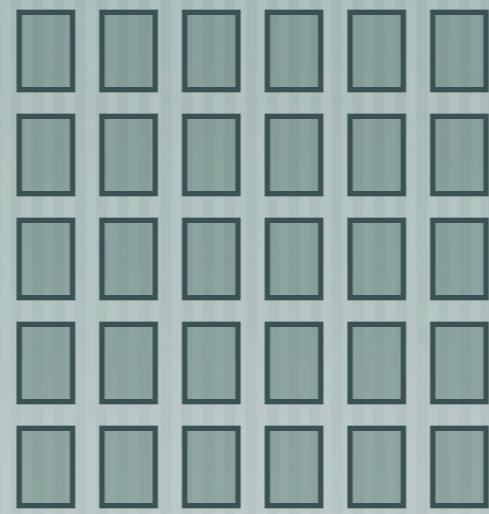
```
pear channel-discover pear.zero.mq  
pecl install zero.mq/zmq-beta  
echo "extension=zmq.so" > \  
      /etc/php.d/zmq.ini
```

<http://github.com/zeromq/zeromq2>

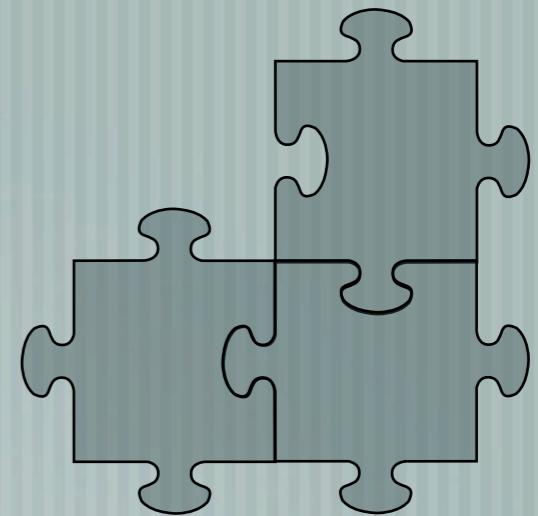
messaging



atomic



string

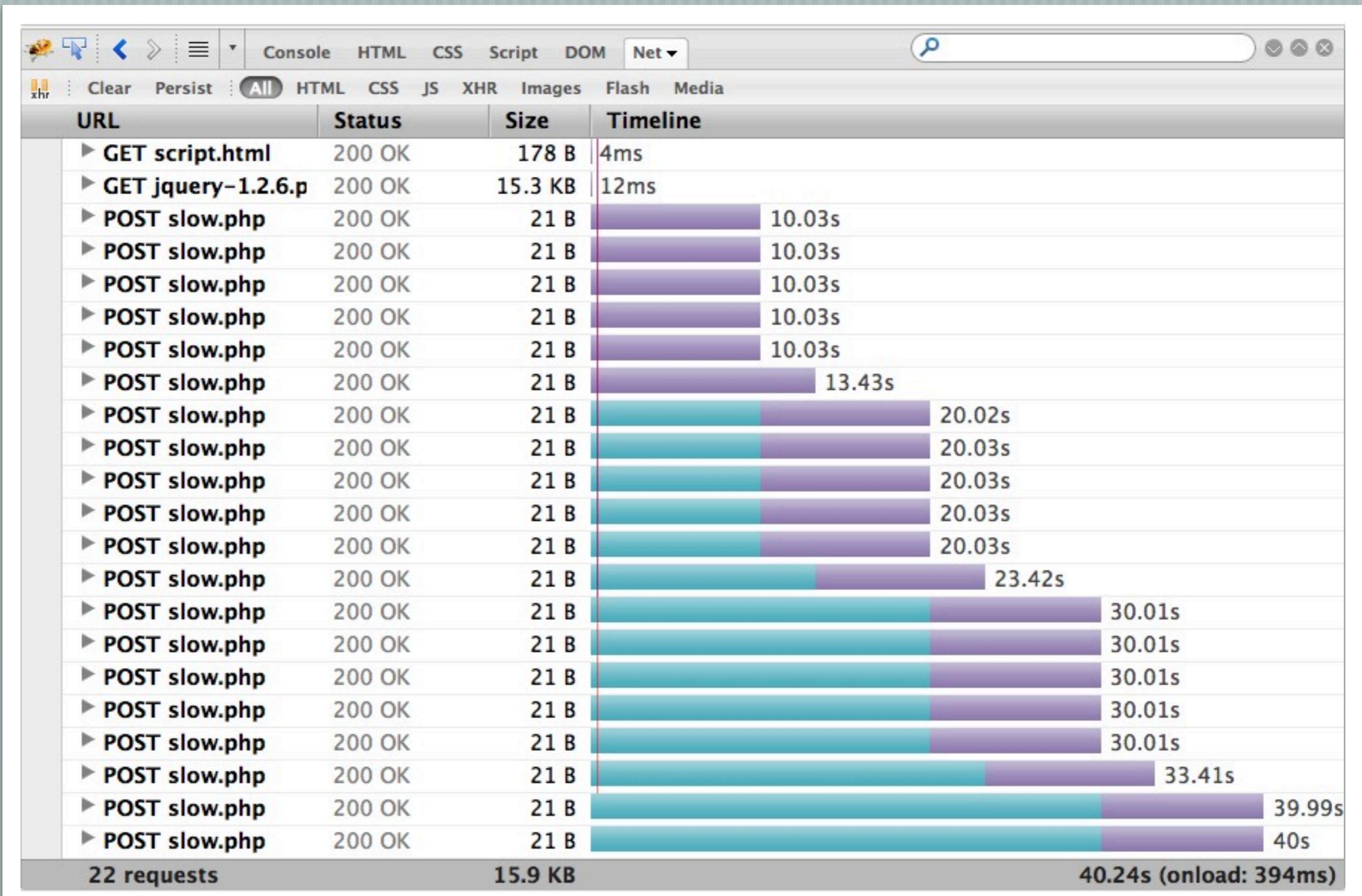


multipart

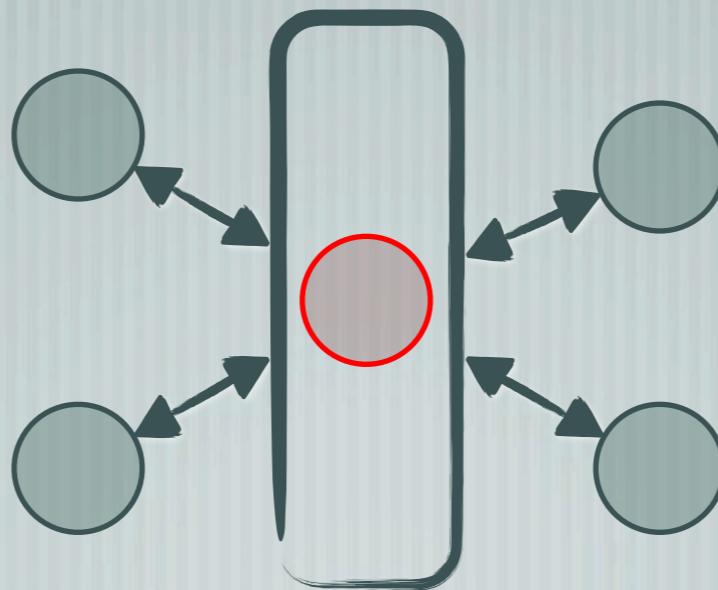
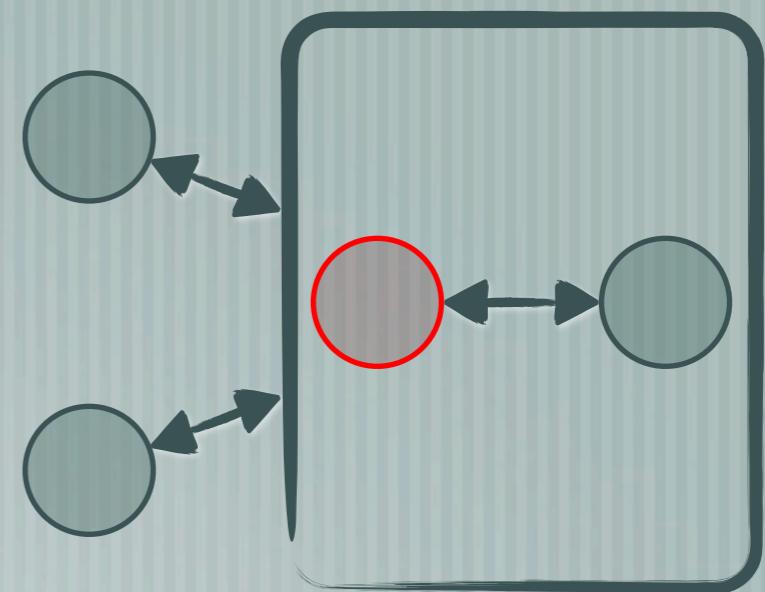
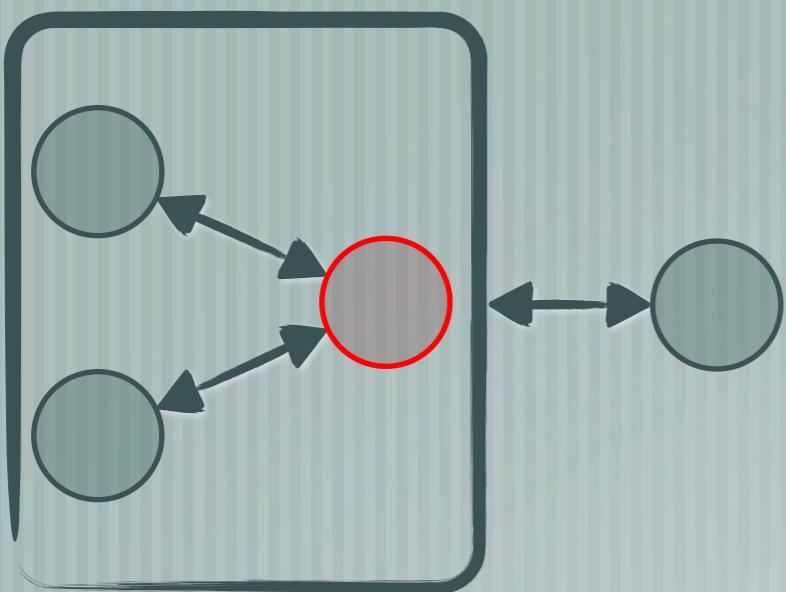


Post Office Image: <http://www.flickr.com/photos/10804218@N00/4315282973>

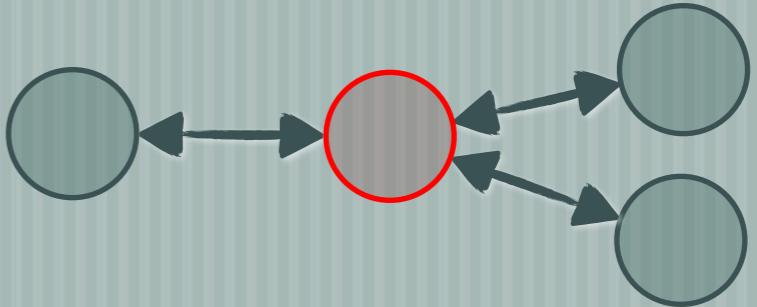
Post Box Image: <http://www.flickr.com/photos/kenjonbro/3027166169>



queue



queue

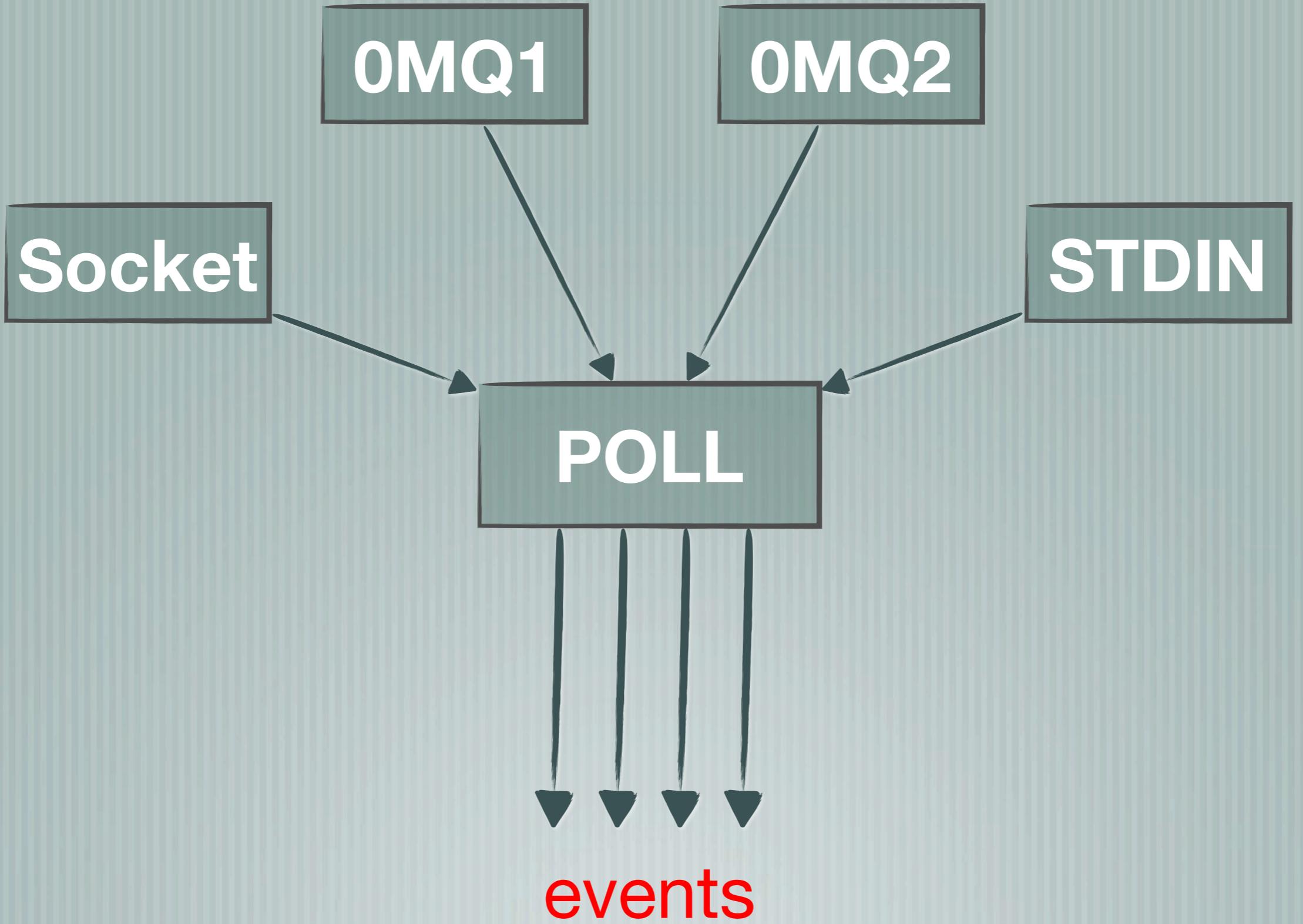


```
$ctx      = new ZMQContext();
$front   = $ctx->getSocket(ZMQ::SOCKET_XREP);
$back    = $ctx->getSocket(ZMQ::SOCKET_XREQ);
$front->bind('tcp://*:5454');
$back->bind ('tcp://*:5455');
```

```
$poll     = new ZMQPoll();
$poll->add($front, ZMQ::POLL_IN);
$poll->add($back, ZMQ::POLL_IN);
$read    = $write = array();
$snd     = ZMQ::MODE SNDMORE;
$rcv     = ZMQ::SOCKOPT RCVMORE;
```

queue.php

```
while(true) {
    $events = $poll->poll($read, $write);
    foreach($read as $socket) {
        if($socket === $front) {
            do {
                $msg   = $front->recv();
                $more = $front->getSockOpt($rcv);
                $back->send($msg, $more ? $snd:0);
            } while($more);
        } else if($socket === $back) {
            do {
                $msg   = $back->recv();
                $more = $back->getSockOpt($rcv);
                $front->send($msg, $more ? $snd:0);
            } while($more);
        }
    }
}
```



```
$ctx = new ZMQContext();
$sock = $ctx->getSocket(ZMQ::SOCKET_PULL);
$sock->bind("tcp://*:5555");
$fh = fopen("php://stdin", 'r');

$poll = new ZMQPoll();
$poll->add($sock, ZMQ::POLL_IN);
$poll->add($fh, ZMQ::POLL_IN);

while(true) {
    $events = $poll->poll($read, $write);
    if($read[0] === $sock) {
        echo "ZMQ: ", $read[0]->recv();
    } else {
        echo "STDIN: ", fgets($read[0]);
    }
}
```

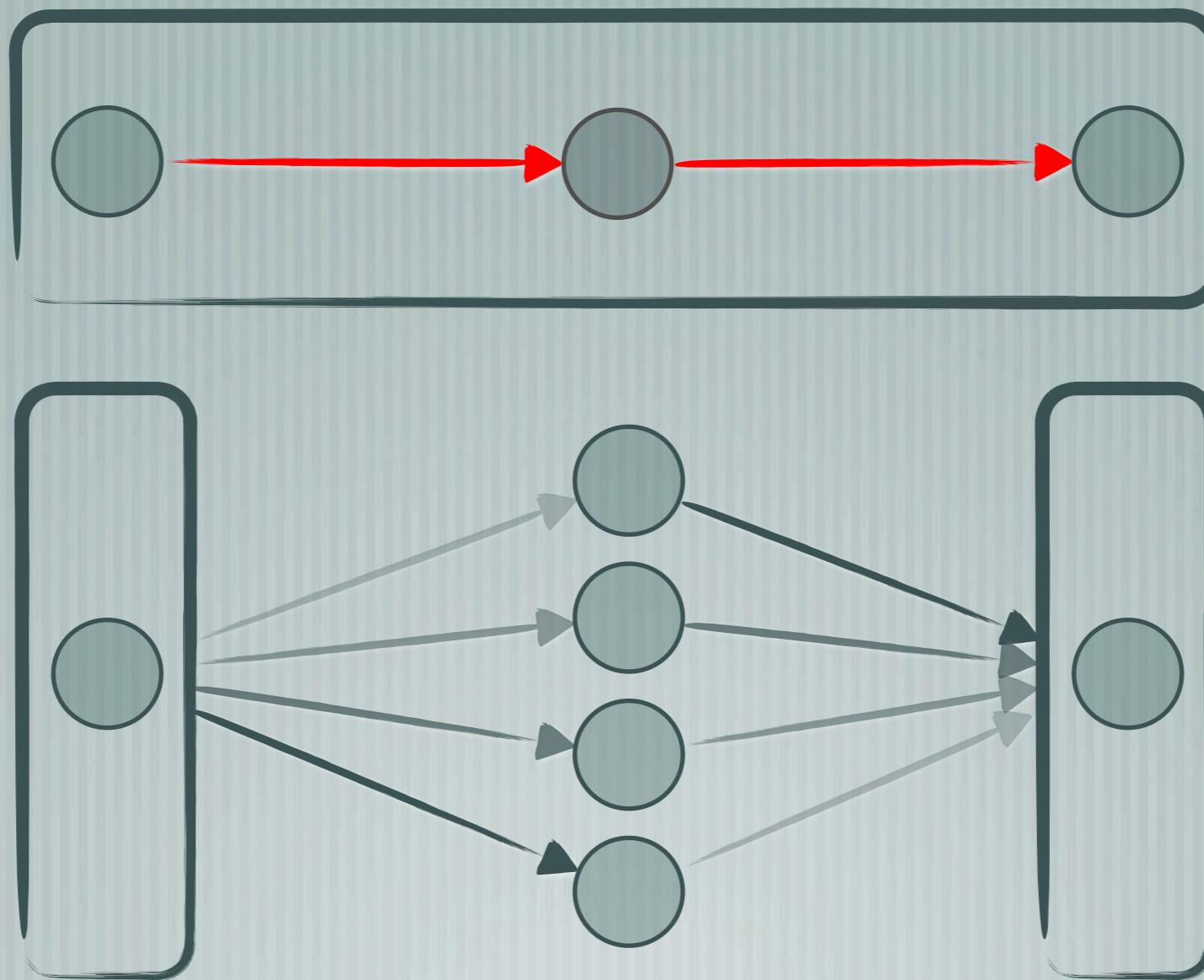
poll.php

stable / unstable

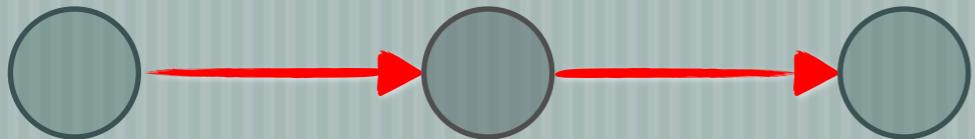


Image: <http://www.flickr.com/photos/pelican/235461339/>

pipeline



pipeline



```
define("NUM_WORKERS", 10);
for($i = 0; $i < NUM_WORKERS; $i++) {
    if(pcntl_fork() == 0) {
        `php work.php`;
        exit;
    }
}
```

controller.php

```
$ctx = new ZMQContext();
$work = $ctx->getSocket(ZMQ::SOCKET_PUSH);
$ctrl = $ctx->getSocket(ZMQ::SOCKET_PUSH);
$work->setSockOpt(ZMQ::SOCKOPT_HWM, 10);
$ctrl->setSockOpt(ZMQ::SOCKOPT_HWM, 1);
```

```
$work->bind("ipc:///tmp/work");
$ctrl->bind("ipc:///tmp/control");
sleep(1);

$fh = fopen('data.txt', 'r');

while($data = fgets($fh)) {
    $work->send($data);
}

for($i = 0; $i < NUM_WORKERS+1; $i++) {
    $ctrl->send("END");
}
```

work.php

```
$ctx = new ZMQContext();
$work = $ctx->getSocket(ZMQ::SOCKET_PULL);
$work->setSockOpt(ZMQ::SOCKOPT_HWM, 1);
$work->connect("ipc:///tmp/work");
$sink = $ctx->getSocket(ZMQ::SOCKET_PUSH);
$sink->connect("ipc:///tmp/results");
$ctrl = $ctx->getSocket(ZMQ::SOCKET_PULL);
$ctrl->setSockOpt(ZMQ::SOCKOPT_HWM, 1);
$ctrl->connect("ipc:///tmp/control");

$poll = new ZMQPoll();
$poll->add($work, ZMQ::POLL_IN);
$read = $write = array();
```

```
while(true) {
    $ev = $poll->poll($read, $write, 5000);
    if($ev) {
        $message = $work->recv();
        $sink->send(strlen($message));
    } else {
        try {
            if($ctrl->recv(ZMQ:::MODE_NOBLOCK)) {
                exit();
            }
        } catch(ZMQException $e) {
            // noop
        }
    }
}
```

sink.php

```
$ctx = new ZMQContext();

$res = $ctx->getSocket(ZMQ::SOCKET_PULL);
$res->bind("ipc:///tmp/results");

$ctrl = $ctx->getSocket(ZMQ::SOCKET_PULL);
$ctrl->setSockOpt(ZMQ::SOCKOPT_HWM, 1);
$ctrl->connect("ipc:///tmp/control");

$poll = new ZMQPoll();
$poll->add($res, ZMQ::POLL_IN);

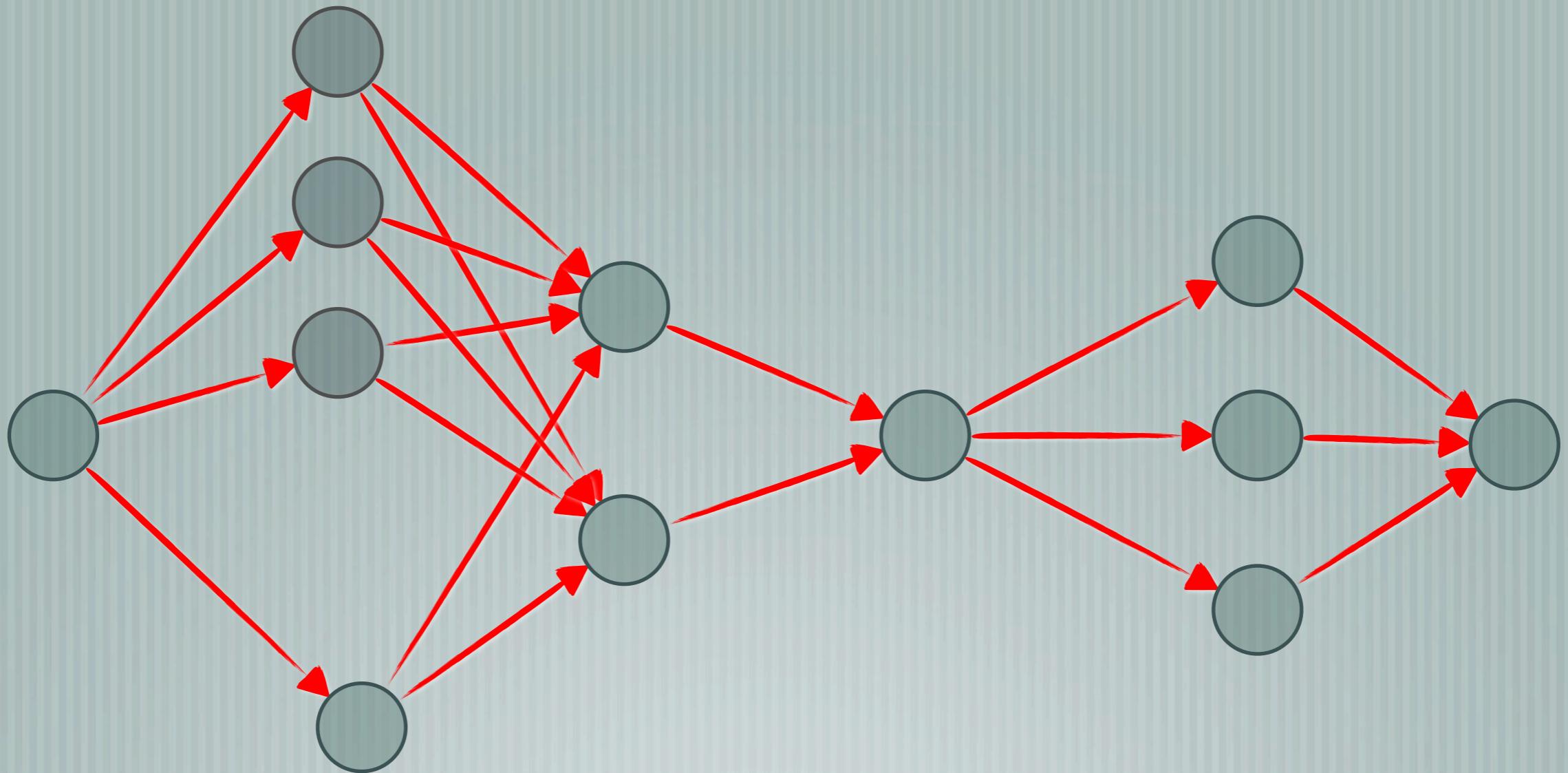
$read = $write = array();
$total = 0;
```

```
while(true) {
    $ev = $poll->poll($read, $write, 10000);
    if($ev) {
        $total += $res->recv();
    } else {
        try {
            if($ctrl->recv(ZMQ::MODE_NOBLOCK)) {
                echo $total, PHP_EOL;
                exit();
            }
        } catch (ZMQException $e) {
            //noop
        }
    }
}
```

```
$ php controller.php
Starting Worker 0
Starting Worker 1
Starting Worker 2
Starting Worker 3
Starting Worker 4
Starting Worker 5
Starting Worker 6
Starting Worker 7
Starting Worker 8
Starting Worker 9
```

```
$ php sink.php
39694
```

filter chain



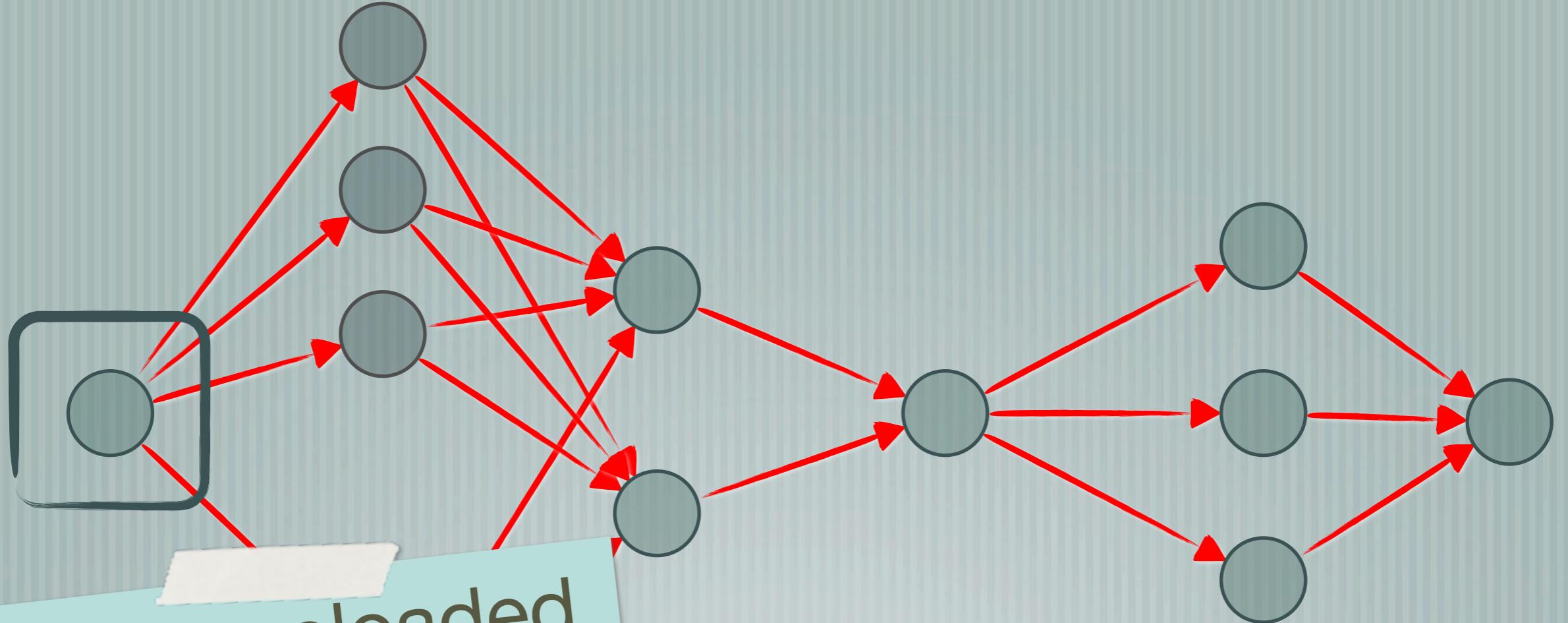
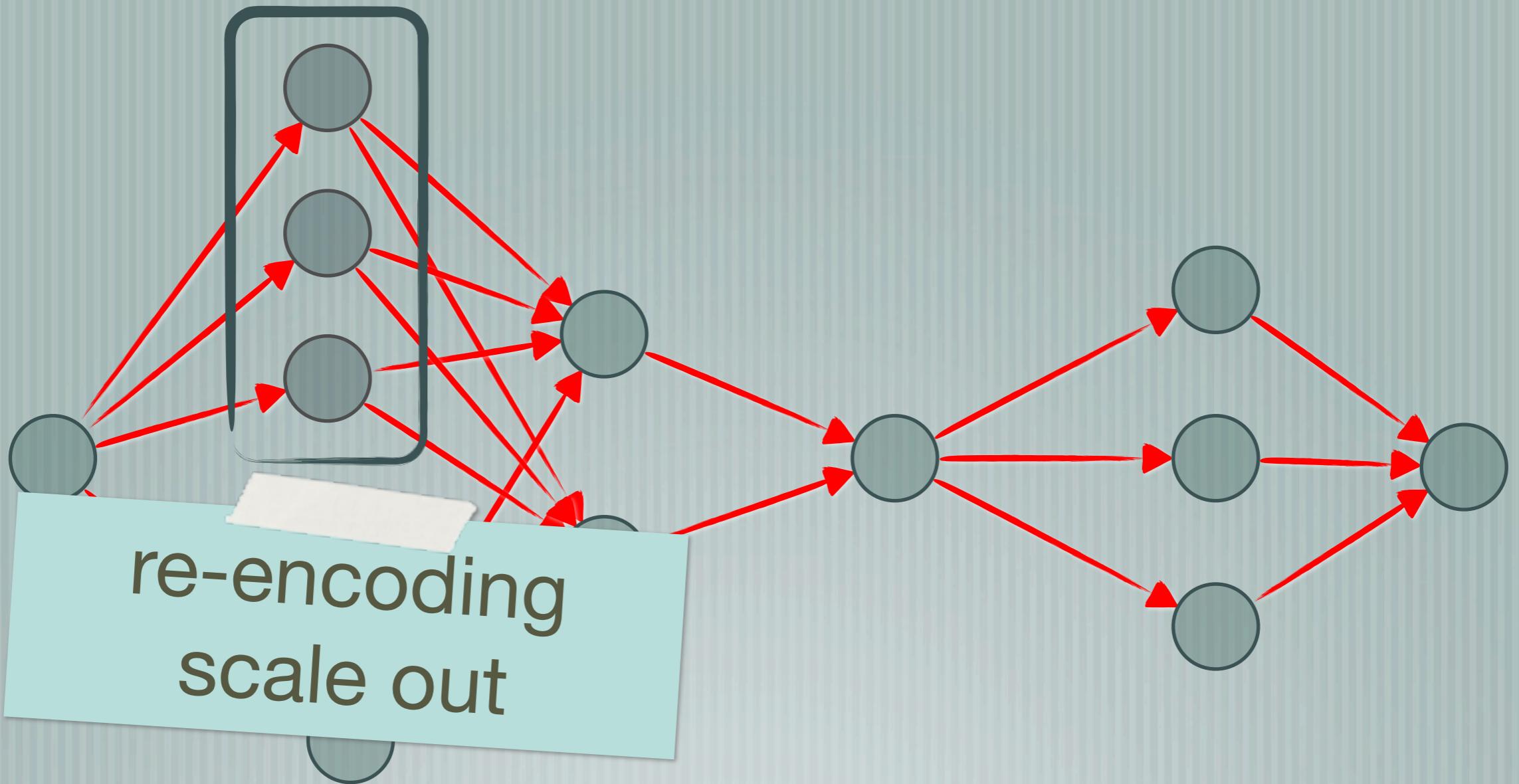
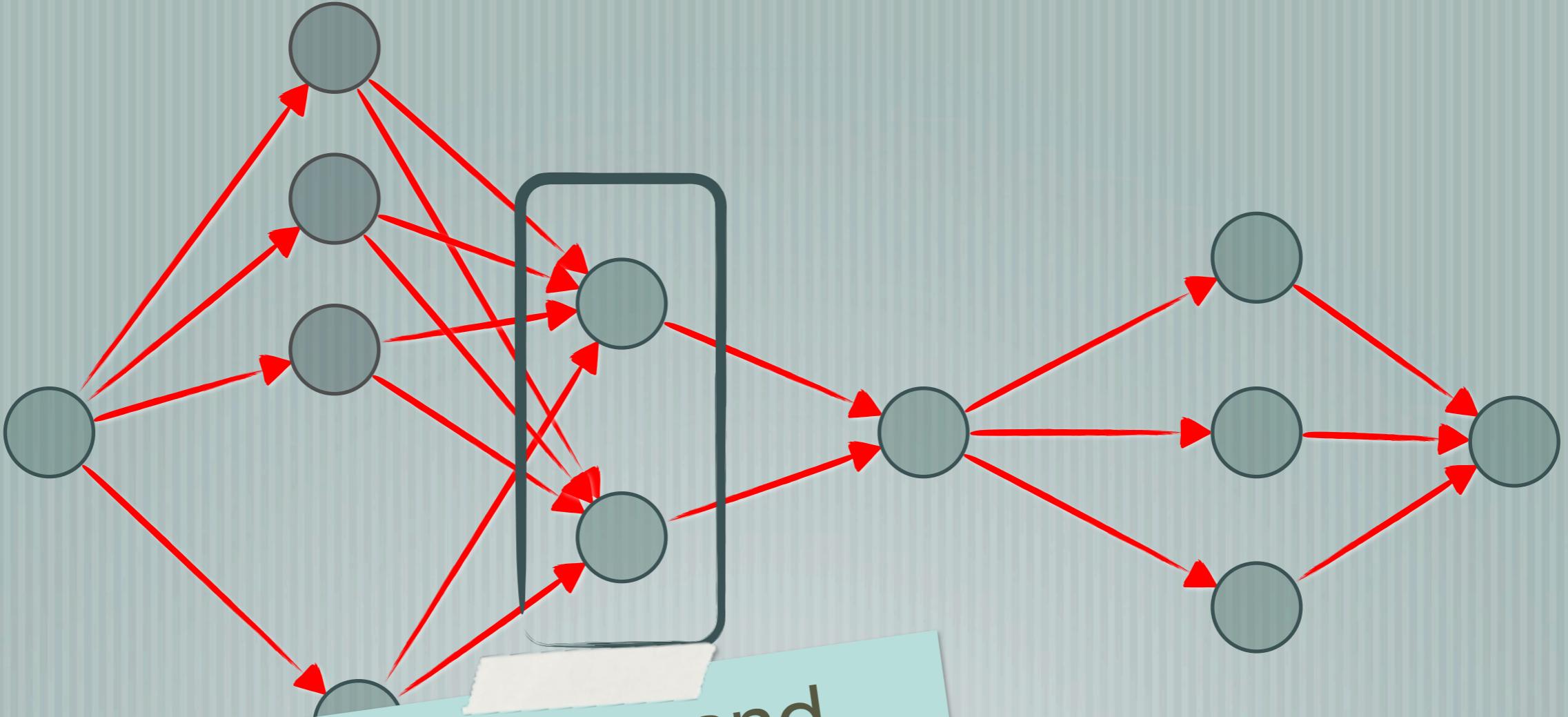
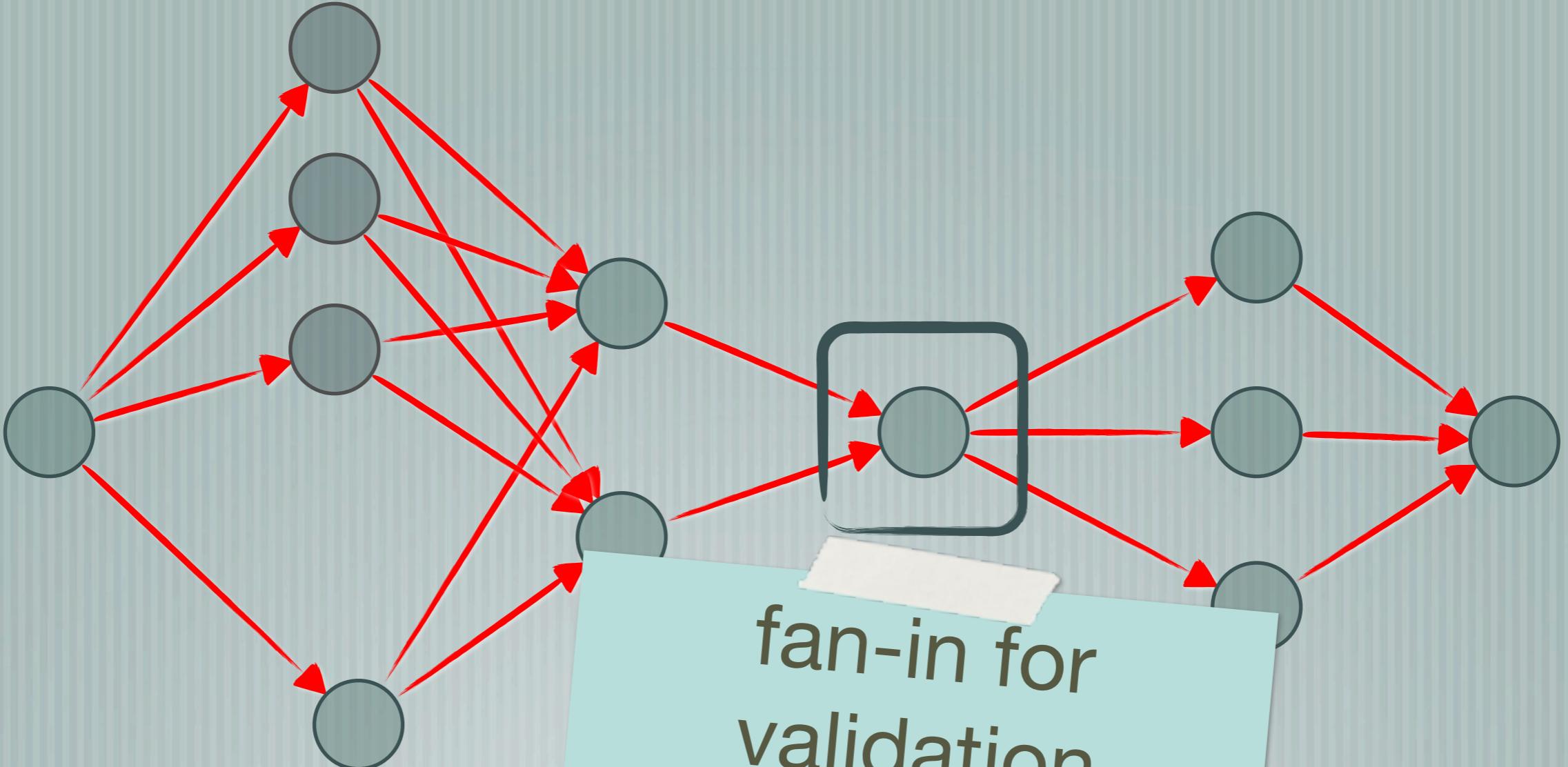


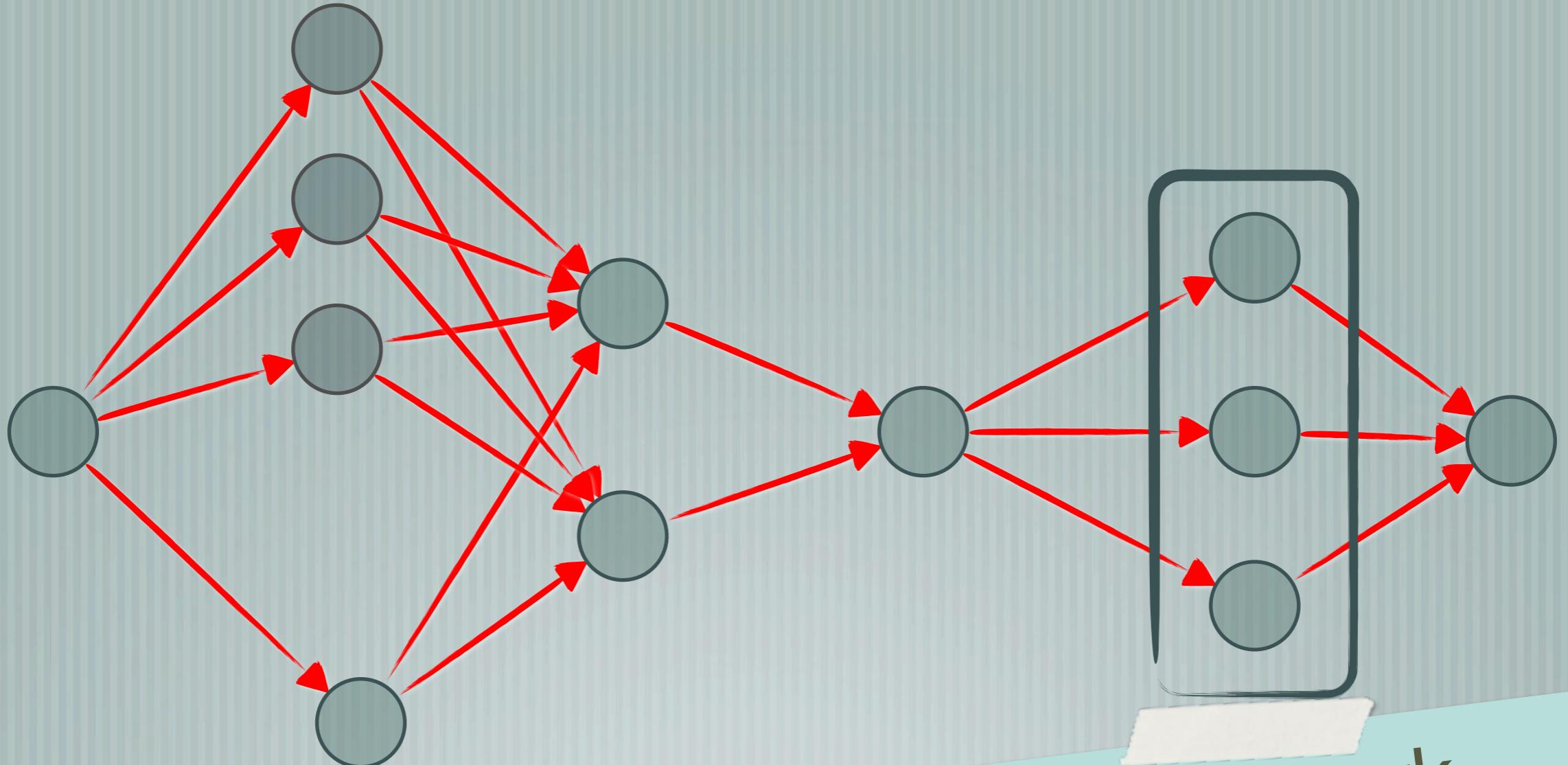
image uploaded
for processing





resizing and
quantizing





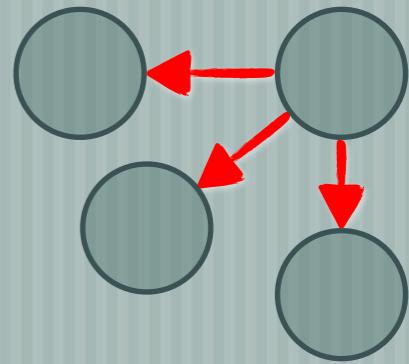
watermark

pub/sub



Image: <http://www.flickr.com/photos/nikonvscanon/4519133003/>

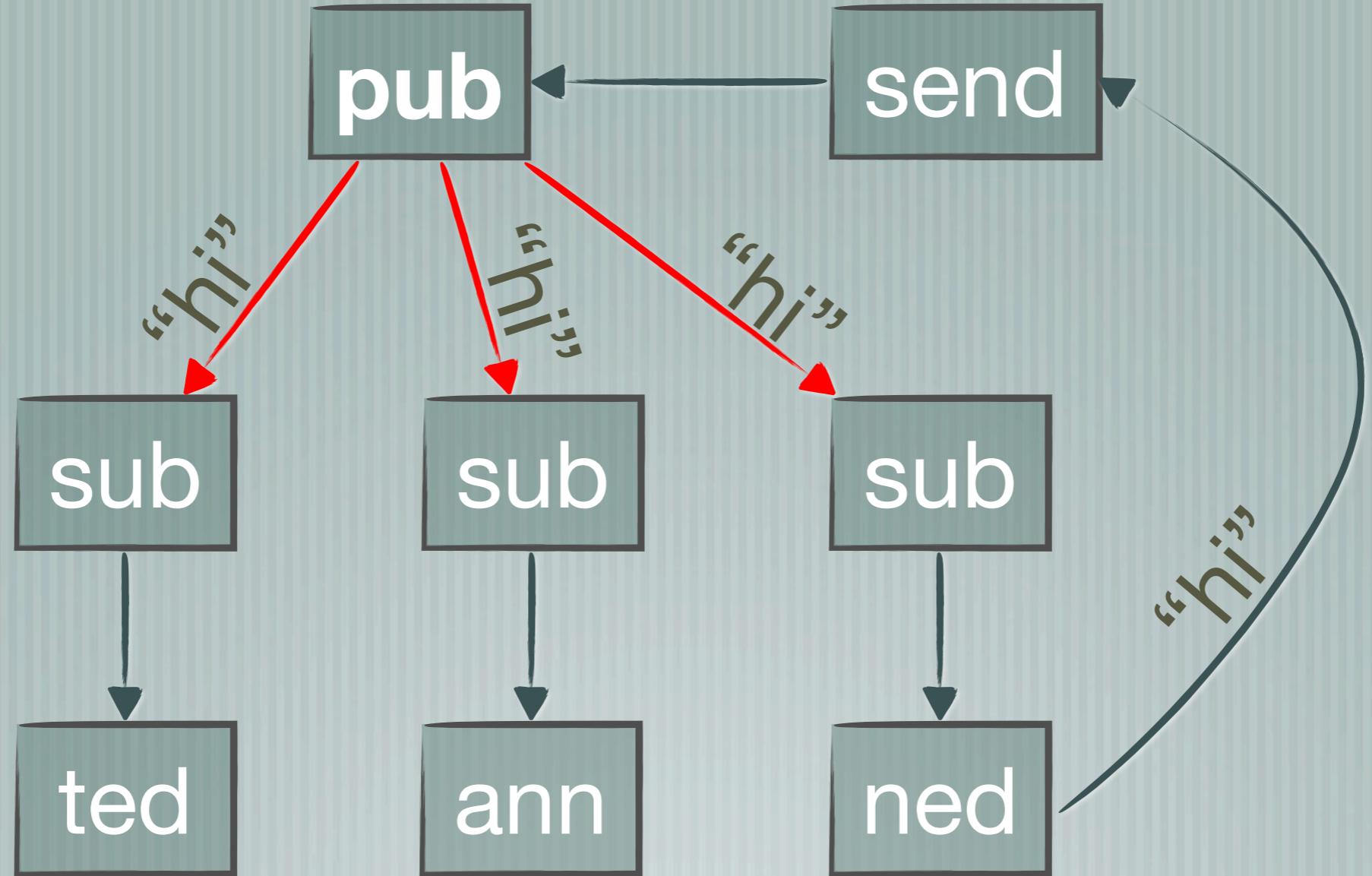
pub/sub



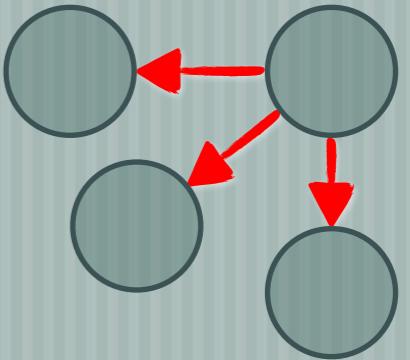
*tony has joined
ian: hello everyone!
tony: hi!*

*ian has joined
tony has joined
ian: hello everyone!
tony: hi!*

Send



pub/sub



```
$ctx = new ZMQContext();
$pub = $ctx->getSocket(ZMQ::SOCKET_PUB);
$pub->bind('tcp://*:5566');
$pull = $ctx->getSocket(ZMQ::SOCKET_PULL);
$pull->bind('tcp://*:5567');

while(true) {
    $message = $pull->recv();
    $pub->send($message);
}
```

server.php

```
$name = htmlspecialchars($_POST['name']);
$msg = htmlspecialchars($_POST['message']);

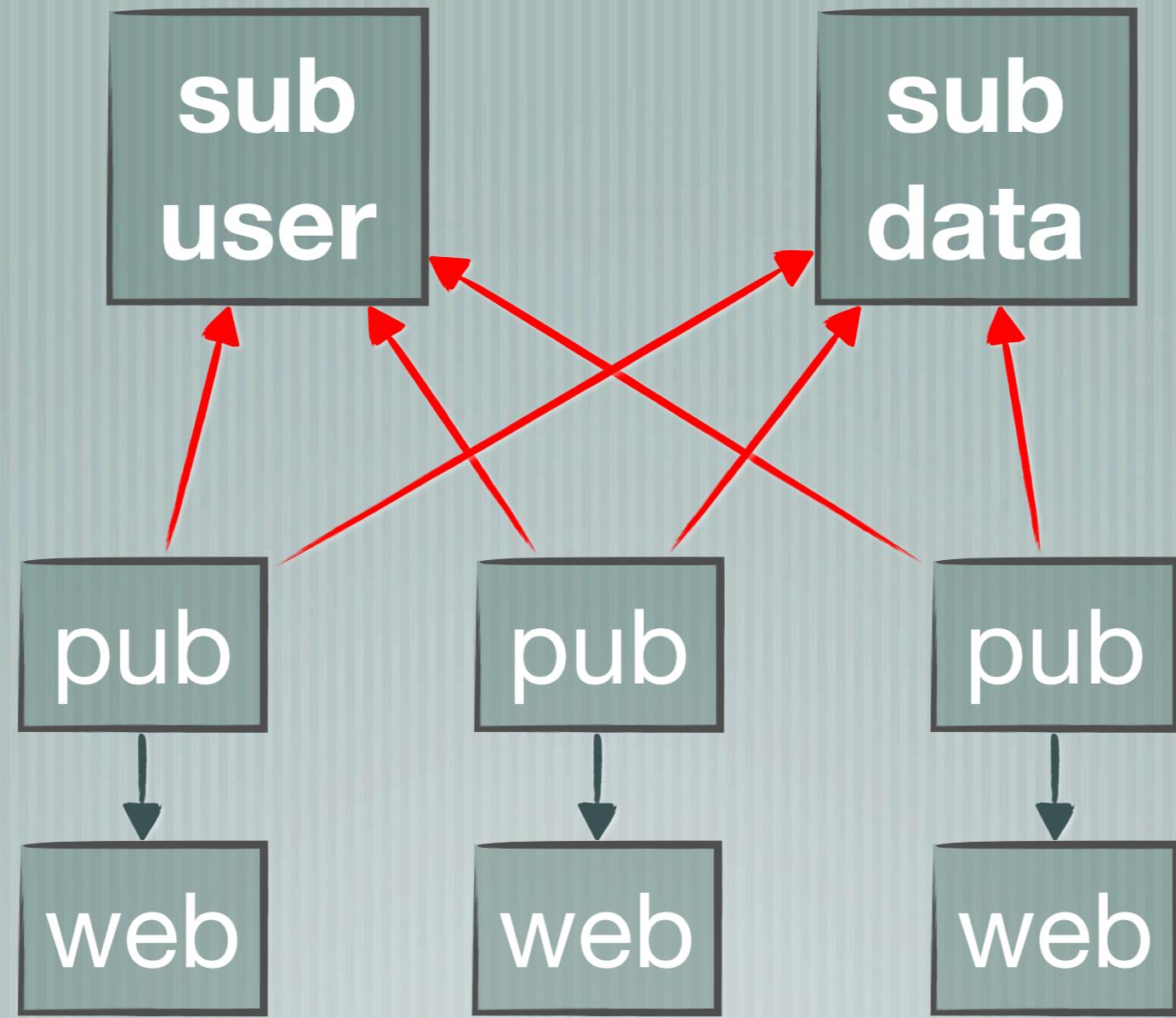
$cxt = new ZMQContext();
$send = $cxt->getSocket(ZMQ::SOCKET_PUSH);
$send->connect('tcp://localhost:5567');

if ($msg == 'm:joined') {
    $send->send(
        "<em>" . $name . " has joined</em>");
} else {
    $send->send($name . ':' . $msg);
}
```

send.php

```
$ctx = new ZMQContext();
$sub = $ctx->getSocket(ZMQ::SOCKET_SUB);
$sub->setSockOpt(ZMQ::SOCKOPT_SUBSCRIBE, '');
$sub->connect('tcp://localhost:5566');
$poll = new ZMQPoll();
$poll->add($sub, ZMQ::POLL_IN);
$read = $wri = array();
while(true) {
    $ev = $poll->poll($read, $wri, 5000000);
    if($ev > 0) {
        echo "<script type='text/javascript'>
                parent.updateChat('";
        echo $sub->recv() . "');</script>";
    }
    ob_flush();
    flush();
}
```

chat.php



```
$ctx = new ZMQContext();
$socket = $ctx->getSocket(ZMQ::SOCKET_PUB);
$socket->connect("ipc:///tmp/usercache");
$socket->connect("ipc:///tmp/datacache");
$type = array('users', 'data');

while(true) {
    $socket->send($type[array_rand($type)],
                  ZMQ::MODE SNDMORE);
    $socket->send(rand(0, 12));
    sleep(rand(0, 3));
}
```

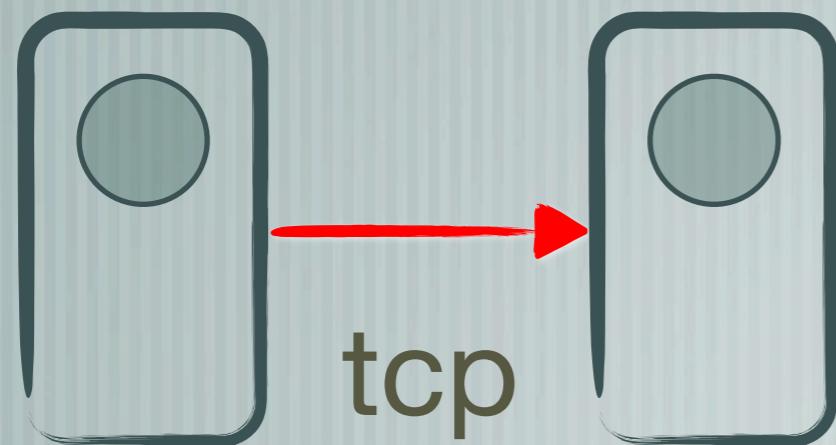
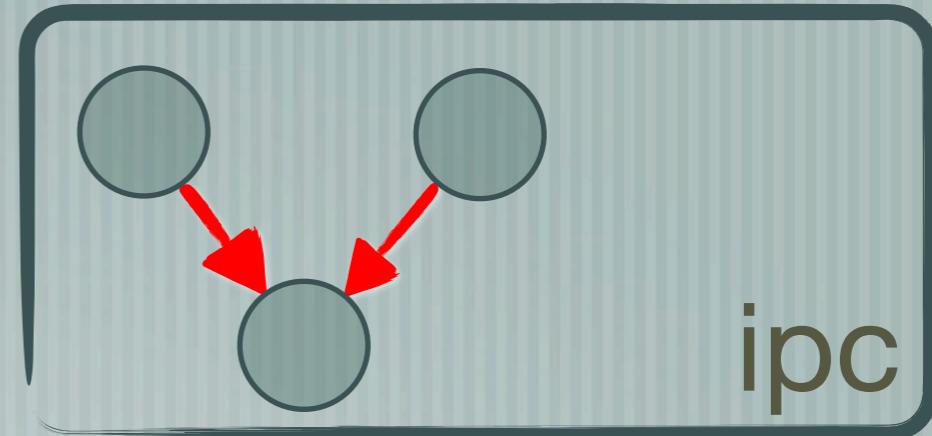
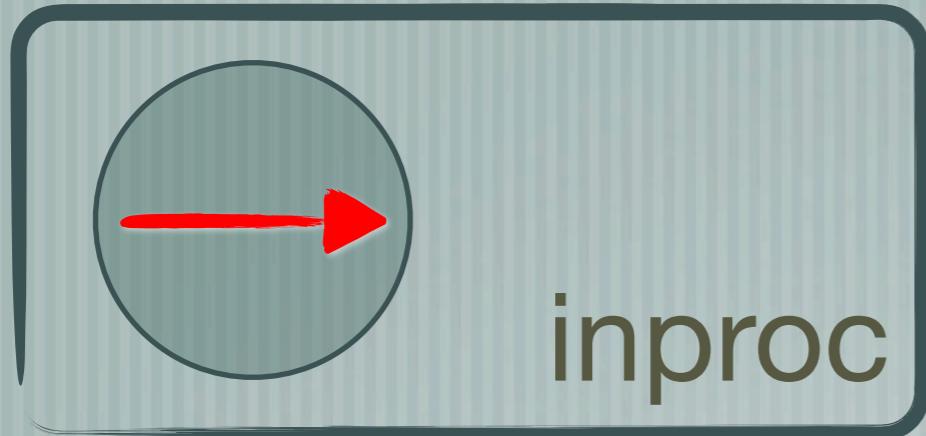
cache.php

```
$ctx = new ZMQContext();
$socket = $ctx->getSocket(ZMQ::SOCKET_SUB);
$socket->setSockOpt(
    ZMQ::SOCKOPT_SUBSCRIBE, "users");
$socket->bind("ipc:///tmp/usercache");

while(true) {
    $cache = $socket->recv();
    $request = $socket->recv();
    echo "Clearing $cache $request\n";
}
```

userlistener.php

types of transport



Realtime results for Oostenrijk



Xhaadiyahsiraj So we've got a TT that's says 'O
10 minutes ago via Mobile Web



ivotju RT @Mike_e: RT @ivotju: Over 2 weke
oostenrijk :) ~~ beter ga je » ja ik ben al we
11 minutes ago via ÜberTwitter



Jordytwello Nederland oostenrijk! Top pl
<http://yfrog.com/h6a8mwj>
11 minutes ago via Twitter for BlackBerry®



VoetbalAlerts #vi :: Van Bommel: 'Oos
gespeeld' #mlalerts: Het Nederlands el
Oost... <http://bit.ly/eNLEPp>
11 minutes ago via twitterfeed



MickvdWijgerd Was kkr hard #nl-o
11 minutes ago via Echofon



Gjoomrss #nrc Oranje verslaat O
Nederlands elftal heeft in Eindh
Oostenrijk... <http://bit.ly/hJ1bKk>
11 minutes ago via twitterfeed



123sport Oranje wint door tv
Nederland heeft in Eindhoven
Oostenrijk met ... <http://bit>
12 minutes ago via twitterfeed



RobNURD me vader wilt v
overhaal kan ik mee.
10 minutes ago via web



Stuart Axon

Ian Barber Laura Newton Percy Cummings may not have been real, but all
the same heres a short story based on a similar scenario:
<http://mybirdie.ca/files/a97da13bf437668a0151b78c370340bd-13180.php>

A Love Story | St. Albert's Place On The Web
mybirdie.ca

22 December 2010 at 00:06 · View post · Remove tag



Ian Barber was tagged in Paul Matthews's album.



Ian's leaving do
5 new photos

13 December 2010 at 08:38

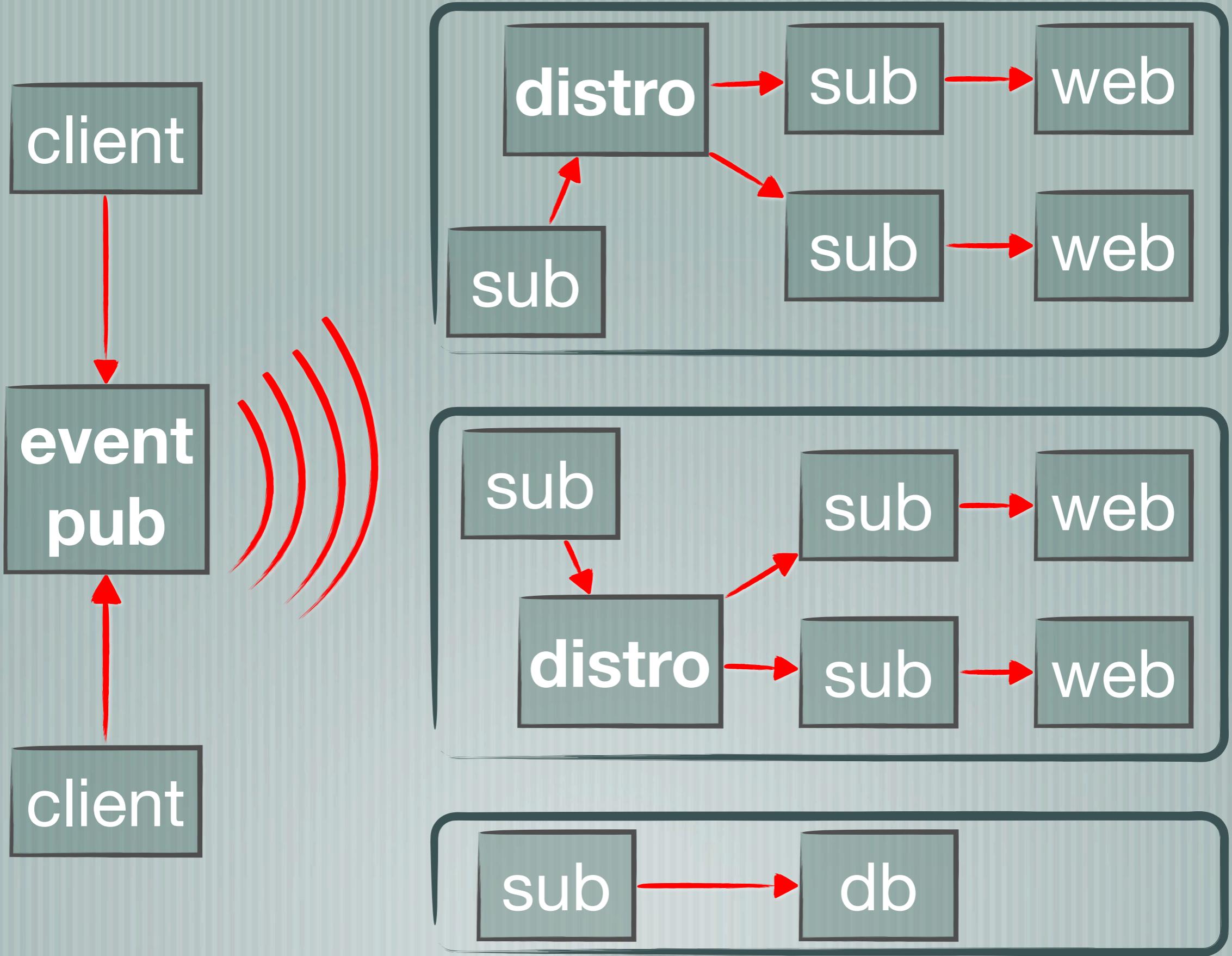
RECENT ACTIVITY

What are you up to this... on Sam Harper's Wall. - See friendship
Ian is now friends with Paul Matthews and 8 other people.



RECENT ACTIVITY

Your pose upsets me Paul, does... on Paul Matthews's photo.
Jo's response to me expressing... on Paul Matthews's photo.
These are *obviously* not th



```
$ctx = new ZMQContext();
$out = $ctx->getSocket(ZMQ::SOCKET_PUB);
$out->setSockOpt(ZMQ::SOCKOPT_RATE, 10000);
$out->connect("epgm://;239.192.0.1:7601");

$in = $ctx->getSocket(ZMQ::SOCKET_PULL);
$in->bind("tcp://*:6767");

$device = new ZMQDevice(
    ZMQ::DEVICE_FORWARDER, $in, $out);
```

eventhub.php

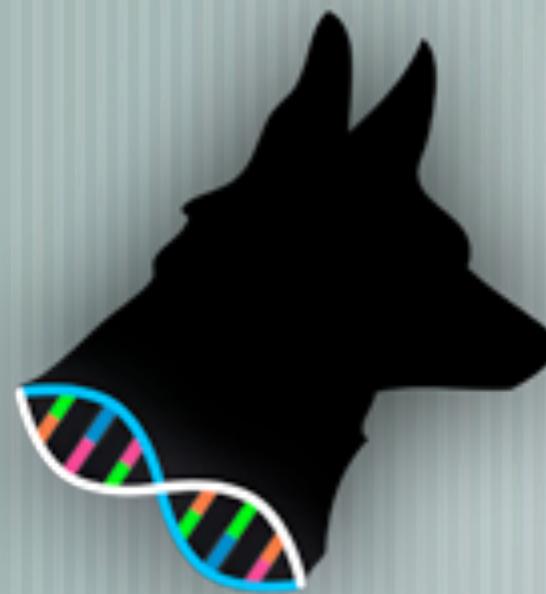
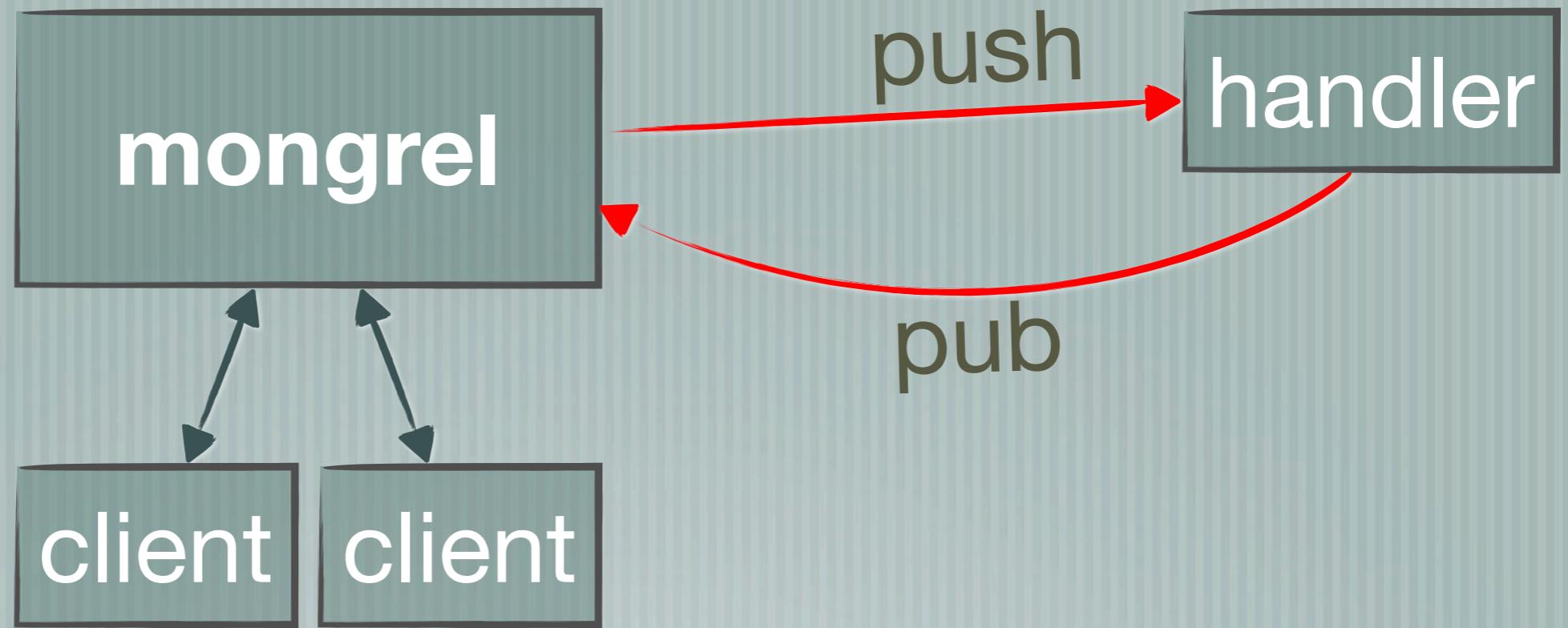
```
$ctx = new ZMQContext();
$in = $ctx->getSocket(ZMQ::SOCKET_SUB);
$in->setSockOpt(ZMQ::SOCKOPT_SUBSCRIBE, '');
$in->setSockOpt(ZMQ::SOCKOPT_RATE, 10000);
$in->connect("epgm://;239.192.0.1:7601");
$out = $ctx->getSocket(ZMQ::SOCKET_PUB);
$out->bind("ipc:///tmp/events");

$device = new ZMQDevice(
    ZMQ::DEVICE_FORWARDER, $in, $out);
```

distro.php

```
$ctx = new ZMQContext();
$in = $ctx->getSocket(ZMQ::SOCKET_SUB);
for($i = 0; $i<100; $i++) {
    $in->setSockOpt(
        ZMQ::SOCKOPT_SUBSCRIBE,
        rand(100000, 999999));
}
$in->connect("ipc:///tmp/events");
$i = 0;
while($i++ < 1000) {
    $who = $in->recv();
    $msg = $in->recv();
    printf("%s %s %s", $who, $msg, PHP_EOL);
}
```

client.php



mongrel 2
<http://mongrel2.org/>

```
$ctx = new ZMQContext();
$in = $ctx->getSocket(ZMQ::SOCKET_PULL);
$in->connect('tcp://localhost:9997');
$out = $ctx->getSocket(ZMQ::SOCKET_PUB);
$out->connect('tcp://localhost:9996');
$http = "HTTP/1.1 200 OK\r\nContent-Length:
%s\r\n\r\n%s";
while(true) {
    $msg = $in->recv();
    list($uuid, $id, $path, $rest) =
        explode(" ", $msg, 4);
    $res = $uuid." ".strlen($id).':'.$id.", ";
    $res .= sprintf($http, 6, "Hello!");
    $out->send($res);
}
```

handler.php

```
simple_handler = Handler(  
    send_spec='tcp://*:9997',  
    send_ident='ab206881-6f49-4276-9db1-1676bfae18b0',  
    recv_spec='tcp://*:9996', recv_ident=''  
)  
main = Server(  
    uuid="9e71cabf-6afb-4ee1-b550-7972245f7e0a",  
    access_log="/logs/access.log",  
    error_log="/logs/error.log",  
    chroot=". /",  
    default_host="general.local",  
    name="example",  
    pid_file="/run/mongre2.pid",  
    port=6767,  
    hosts = [  
        Host(name="general.local",  
              routes={'/test':simple_handler})  
    ]  
)  
settings = {"zeromq.threads": 1}  
servers = [main]
```

```
namespace m2php;  
$id ="82209006-86FF-4982-B5EA-D1E29E55D481";  
$con = new \m2php\Connection($id,  
                            "tcp://127.0.0.1:9997",  
                            "tcp://127.0.0.1:9996");  
$ctx = new \ZMQContext();  
$in = $ctx->getSocket(\ZMQ::SOCKET_SUB);  
$in->setSockOpt(\ZMQ::SOCKOPT_SUBSCRIBE, '' );  
$sub->connect('tcp://localhost:5566');  
$poll = new \ZMQPoll();  
$poll->add($sub, \ZMQ::POLL_IN);  
$poll->add($con->reqs, \ZMQ::POLL_IN);  
$read = $write = $ids = array(); $snd = '';  
$h = "HTTP/1.1 200 OK\r\nContent-Type: text/  
html\r\nTransfer-Encoding: chunked\r\n\r\n";
```

<https://github.com/winks/m2php>

```
while (true) {
    $ev = $poll->poll($read, $write);
    foreach ($read as $r) {
        if ($r === $in) {
            $m = "<script type='text/javascript'>
parent.updateChat('". $in->recv()."');
</script>\r\n";
            $con->send($snd, implode(' ', $ids),
                      sprintf("%x\r\n%$s", strlen($m), $m));
        } else {
            $req = $con->recv();
            $snd = $req->sender;
            if ($req->is_disconnect()) {
                unset($ids[$req->conn_id]);
            } else {
                $ids[$req->conn_id] = $req->conn_id;
                $con->send($snd, $req->conn_id, $h);
            }
        }
    }
}
```

Helpful Links

<http://zero.mq>

<http://zguide.zero.mq>

<http://github.com/ianbarber/ZeroMQ-Talk>

<http://joind.in/talk/view/2523>

thanks!

