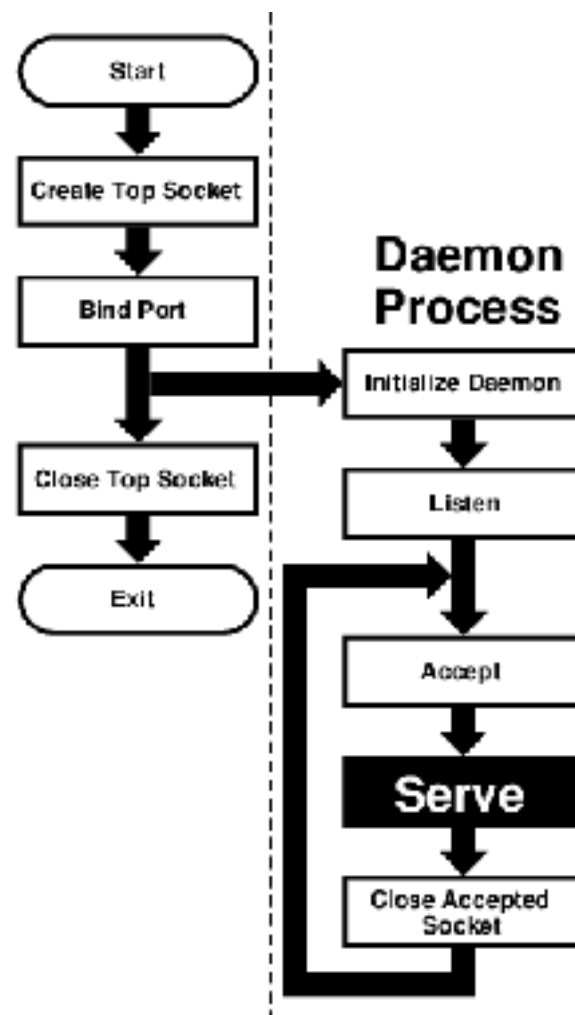


Server schreiben? Das kann doch jeder!





```
int main(int argc, char **argv) {

    if ((fd = socket(AF_INET, SOCK_STREAM, 0)) < 0)
        return 1;

    addr.sin_family = AF_INET;
    addr.sin_addr.s_addr = htonl(INADDR_ANY);
    addr.sin_port = htons(8888);

    if (bind(fd, (struct sockaddr *) &addr,
             sizeof(addr)) < 0)
        return 1;

    if (listen(fd, 5) < 0)
        return 1;

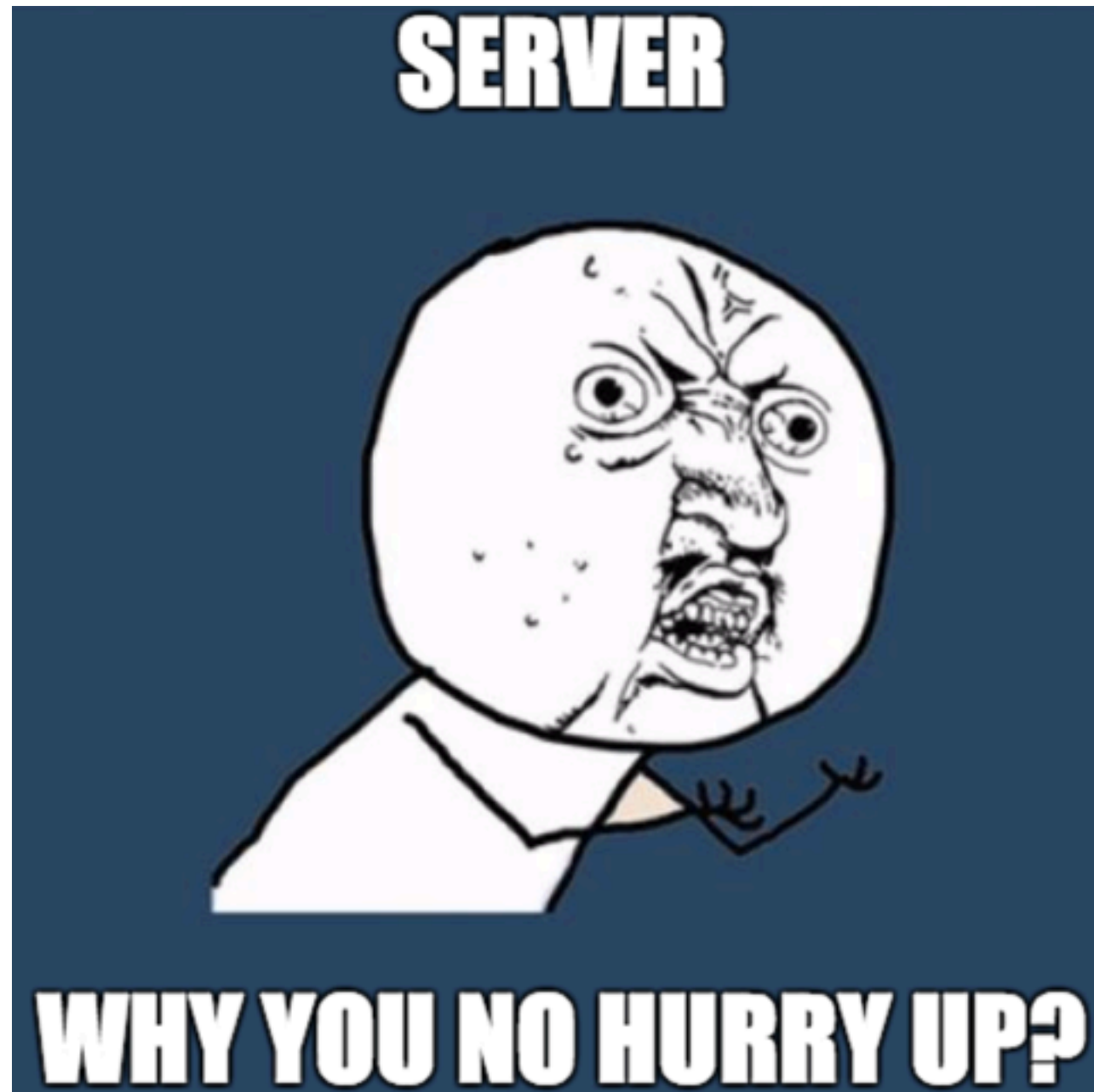
    while (1) {

        if ((cfd = accept(fd, (struct sockaddr *) &clientaddr,
                         &clientlen)) < 0)
            return 1;

        if ((n = read(cfd, buf, 1024)) < 0)
            return 1;

        if ((n = write(cfd, buf, strlen(buf))) < 0)
            return 1;

        close(cfd);
    }
}
```



Lösung? Event-Driven!

```

int main(int argc, char **argv) {

    if ((fd = socket(AF_INET, SOCK_STREAM, 0)) < 0)
        return 1;

    ...

    if (listen(fd, SOMAXCONN) < 0)
        return 1;

    if ((efd = epoll_create1(0)) < 0)
        return 1;

    event.data.fd = sfd;
    event.events = EPOLLIN | EPOLLET;
    if (epoll_ctl (efd, EPOLL_CTL_ADD, sfd, &event) < 0)
        return 1;

    while (1) {

        n = epoll_wait(efd, events, MAXEVENTS, -1);
        for (i = 0; i < n; i++) {
            if (events[i].data.fd == fd) {
                if ((cfd = accept(fd, (struct sockaddr *) &clientaddr,
                                &clientlen)) < 0)
                    return 1;

                event.data.fd = infd;
                event.events = EPOLLIN | EPOLLET;
                epoll_ctl(efd, EPOLL_CTL_ADD, cfd, &event);

            } else {

                // read/write on
                // event.data.fd

            }
        }
    }
}

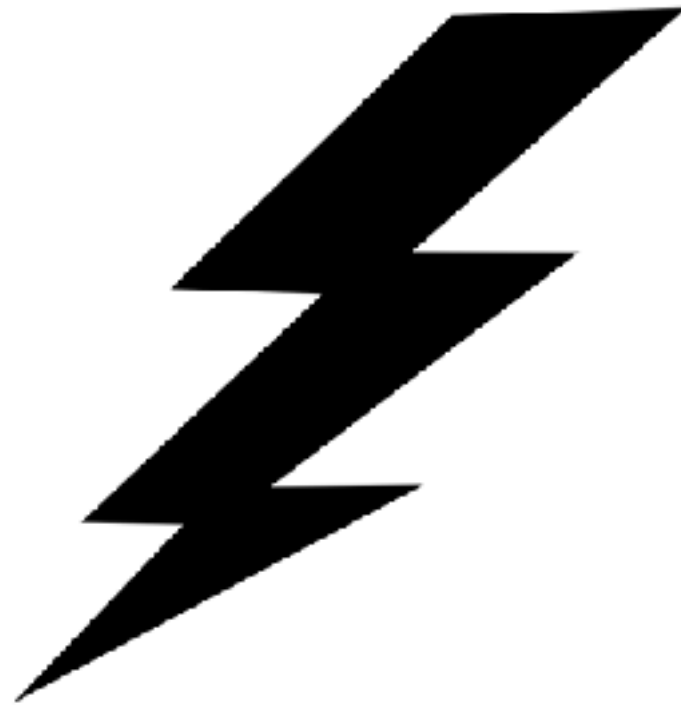
```

# Abstraction!

# Abstraction!

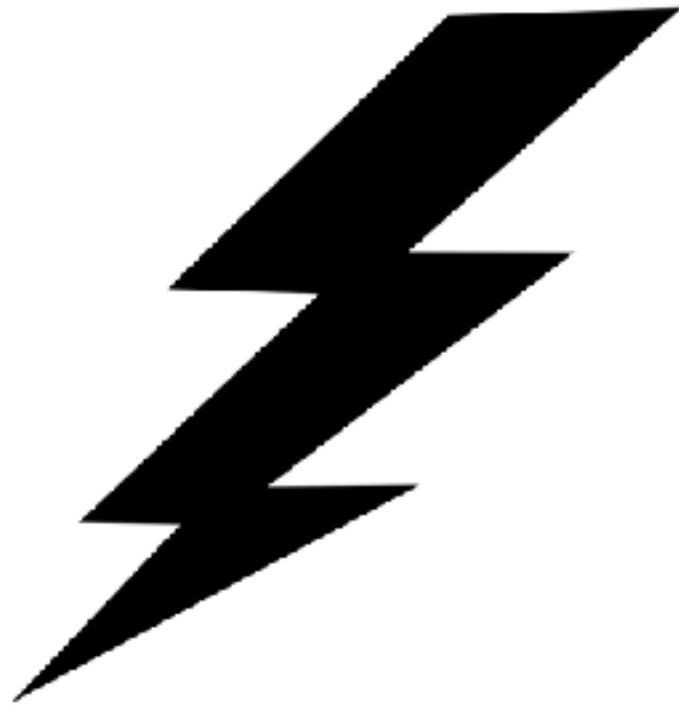
- libevent
- libev





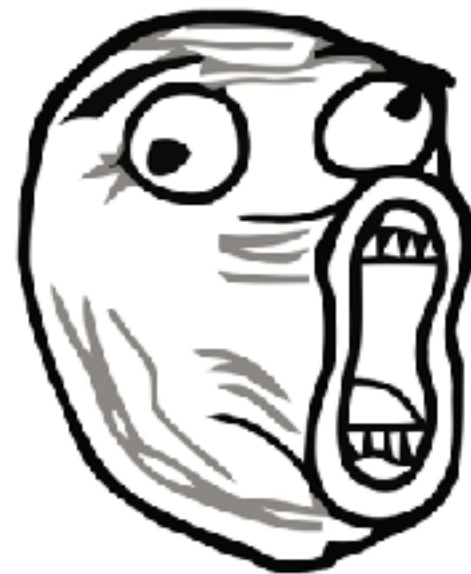
Zeitsprung: 2008

Robert: PHP FTW!



Zeitsprung: 2008

Robert: PHP FTW!





```
<?php
```

```
$v8 = new V8();  
$v8->exec("foo = 123");  
echo $v8->foo;
```



+



**libuv**

=



## node.js FTW!

```
var net = require('net');

var server = net.createServer(function(socket) {
  socket.write('Echo server\r\n');
  socket.pipe(socket);
});

server.listen(8888, '127.0.0.1');
```

# Vorteile node.js?

# Vorteile node.js?

**Atwood's Law:** Any application that *can* be written in JavaScript, *will* eventually be written in JavaScript

- Schnelle Entwicklung
- Riesige Community
- npm!!11elf
- Einfacher Layer auf Kernel

# Bildquelle

WWW