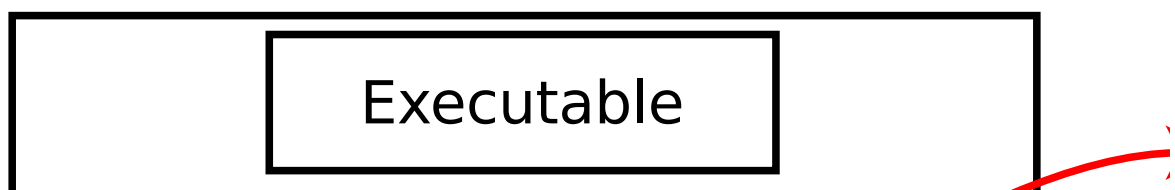



Red arrow: Call  
Blue arrow: Return  
Black arrow: Next

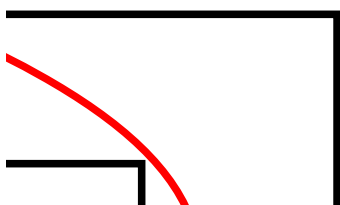
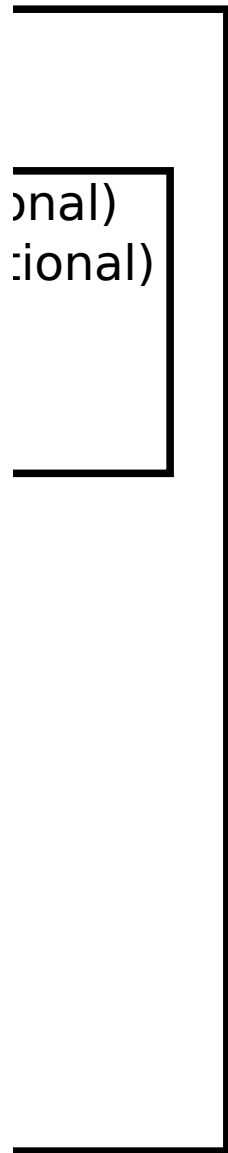


## Modules

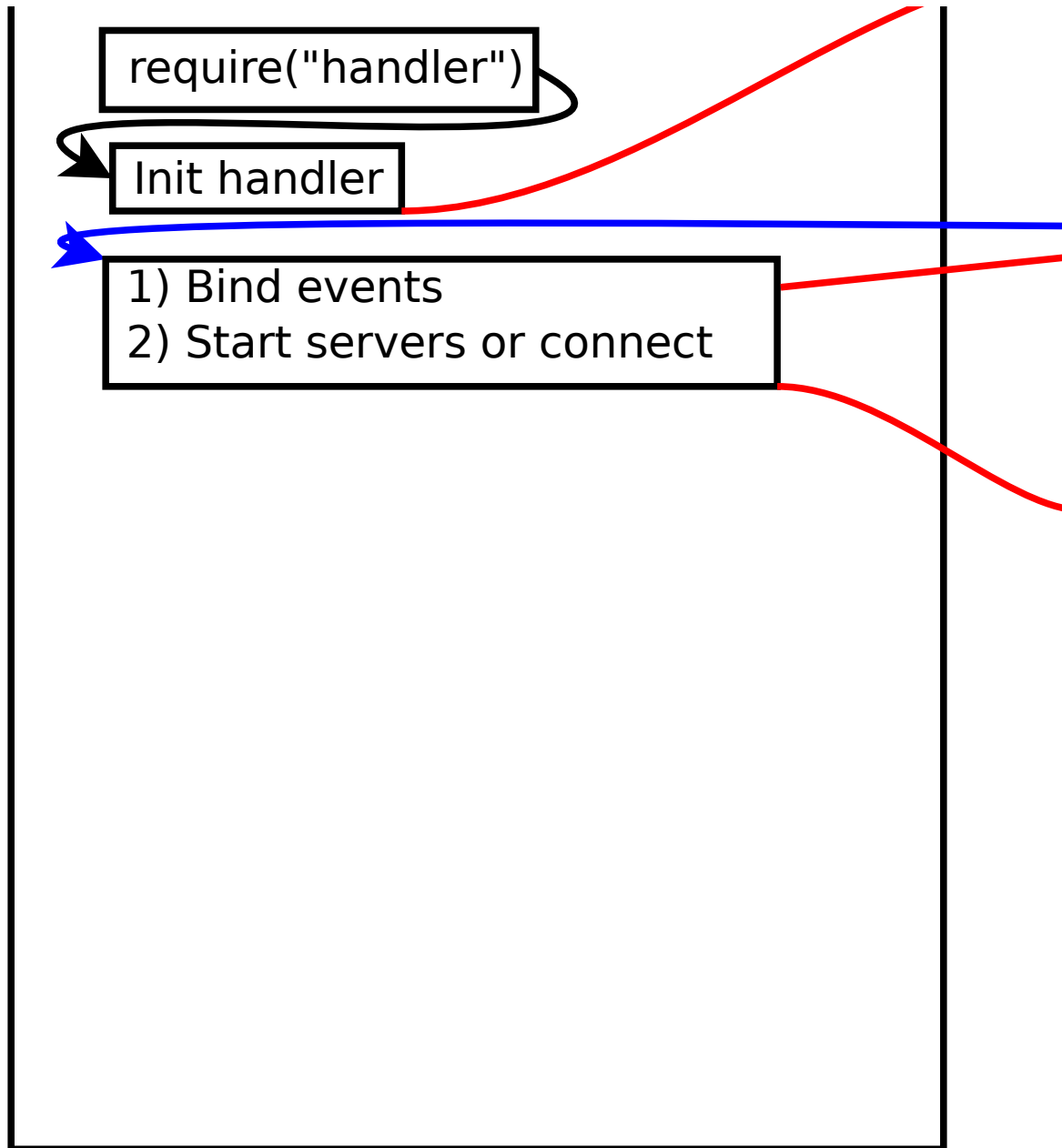
- 
- 1) Request local (per-instance) memory (opti
  - 2) Request global (all instances) memory (opt
  - 3) Register web pages (sort of a web-event?)
  - 4) Register commands (it's an event)
  - 5) Register for other events

## Handler

- 
- 1) Read/parse settings







- 2) Initialize proto handlers (IRC, email, http, ic
- 3) Ready event handling/memory spaces
- 4) Load modules

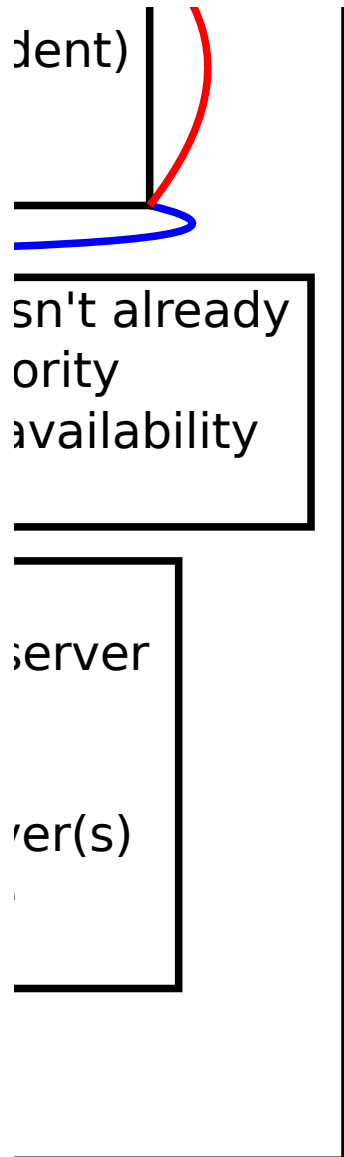


- 1) Bind to request entity (certain IRC, etc.) if is
  - 2) Add function to given event in entity by prio
- Name is stored here too, so that a callback's a
- is togglable without removal.



If requested to in settings do:

- 1) Start email server or connect to an email s
  - 2) Start ident server
  - 3) Start webserver
  - 4) Start IRC server(s) and connect to IRC serv
- Each IRC server and connection is a separate instantiation.



- 1) Connect to gi
- 2) Fire connecte

- 1) Start server li
- 2) Fire server-up
- 3) Fire event for

- 1) Remember ca
- Note that in here

IRC

ven server  
d event when complete

listening on given host/port  
event when complete  
each connection

allback for event  
e there is only one callback per event.









