MESH: MESSAGING DESIGN OVERVIEW

Last updated: 6 January 2011

Communication Layer

port_requestor

Connect to another node's port_assigner and ask for proxy ports.

PULL — IPC BIND

PUSH — IPC CONN

REQ — TCP CONN

communicator

PUSH — IPC CONN

PUSH — IPC CONN

Core of all communications. Route messages passing through this node. Send event messages out. Receive event messages from other nodes. Handle mesh node topology. Connect to other nodes.

PUSH — TCP CONN

x N

starts at *:4205

PUSH — TCP BIND

x N

PULL — IPC BIND

PUSH — IPC CONN

port_assigner

Tell another node's port_requestor which proxy ports to connect to.

PULL — IPC BIND

PUSH — IPC CONN

REP — TCP BIND *:4200

pull_proxy

(1 x inbound TCP)
(N x outbound TCP)

PULL — TCP BIND *:4201

OR

PULL — TCP CONN

Pass inbound traffic from nodes that connect to us on to communicator.

PUSH — IPC CONN

x N

Core Layer

master

PUSH — IPC CONN

Launches communication layer processes and plugins. Filters events from plugins.

PULL — IPC BIND

(plugins)

Each plugin monitors something specific to the box it is running on for an event to happen.

PUSH — IPC CONN

x N

Legend:

→ **Direction** TCP connection is established

→ **Direction** ZMQ Messages flow (over TCP)

→ **Direction** ZMQ Messages flow (interprocess)