



# HTCondor Networking Concepts

# Disclaimers

- › Not about configuration macros
- › Not about host or daemon lookups
- › Not about HTCondor internals

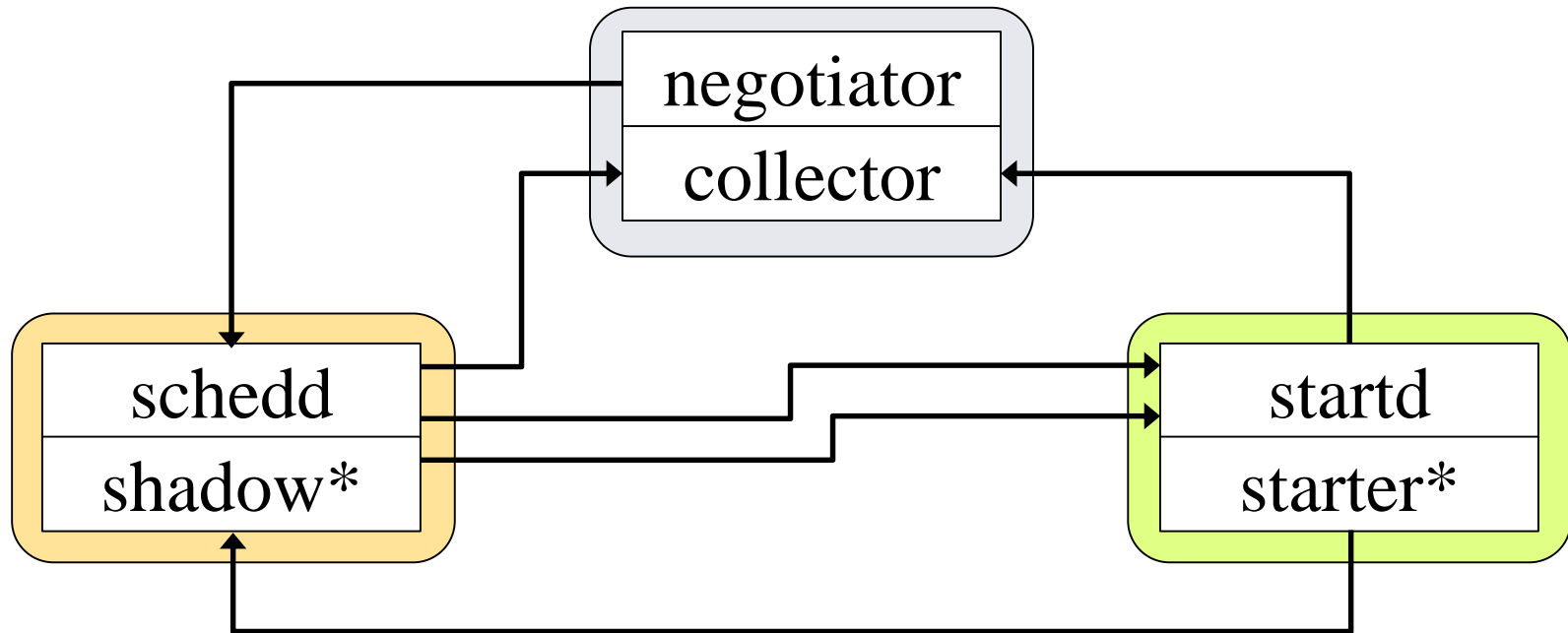
# Asking the Right Questions

- › There will be a quiz at the end
- › Start by reviewing fairy-tale networking
- › ... then add IPv6
- › ... then add schedd firewalls
- › ... then add startd firewalls
- › End by passing the quiz (open-manual)

# Fairy-tale Networking

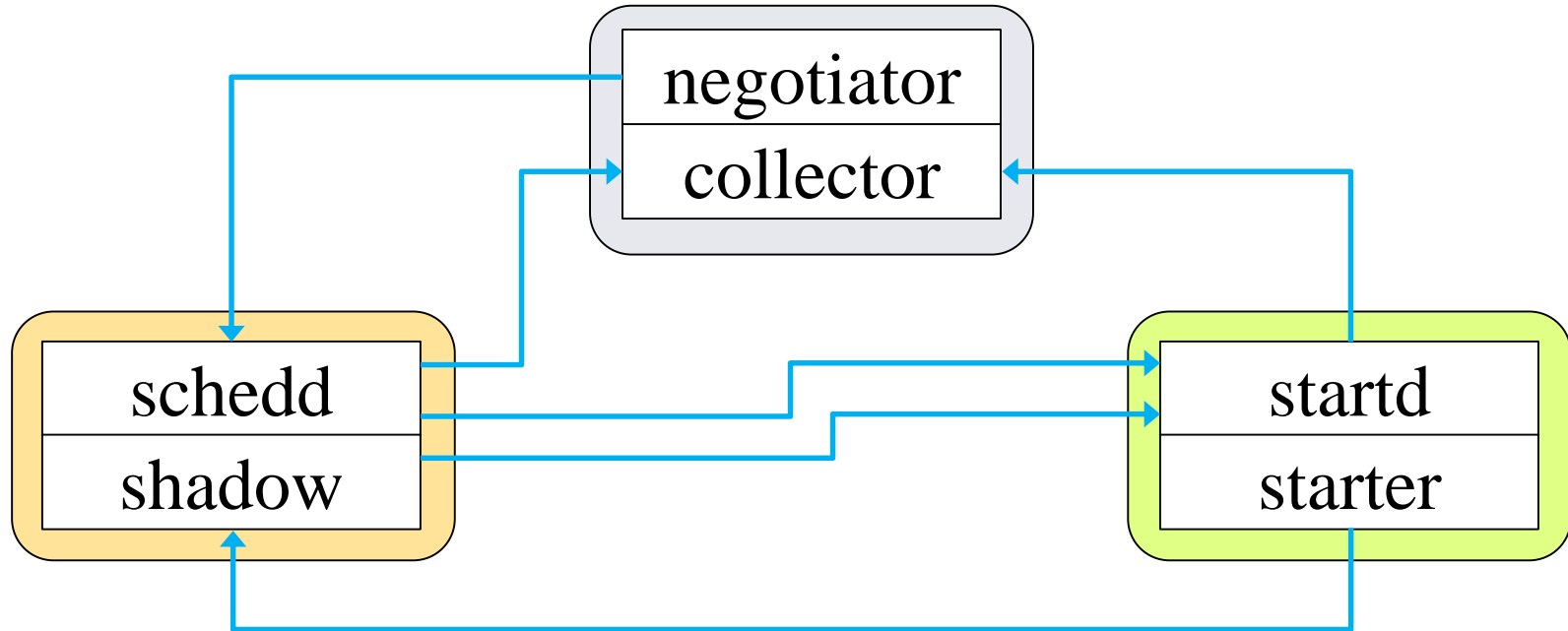
- › Single network protocol
- › All addresses publically routable
- › No firewalls
- › Fewer than ~25k simultaneous running jobs

# Working in a Fairy Tale



\* One shadow, starter per running job

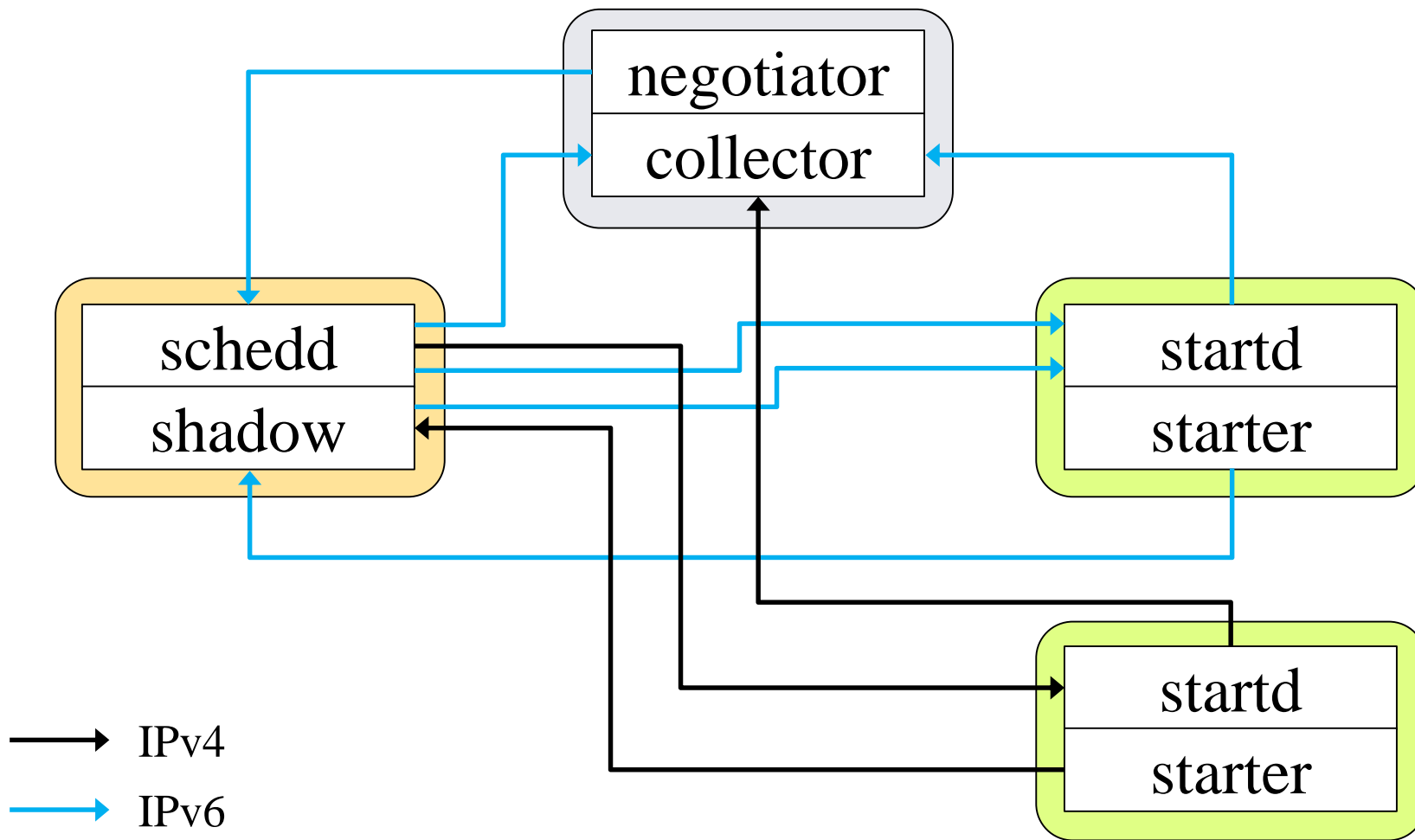
# IPv6



→ IPv4

→ IPv6

# IPv6 + IPv4

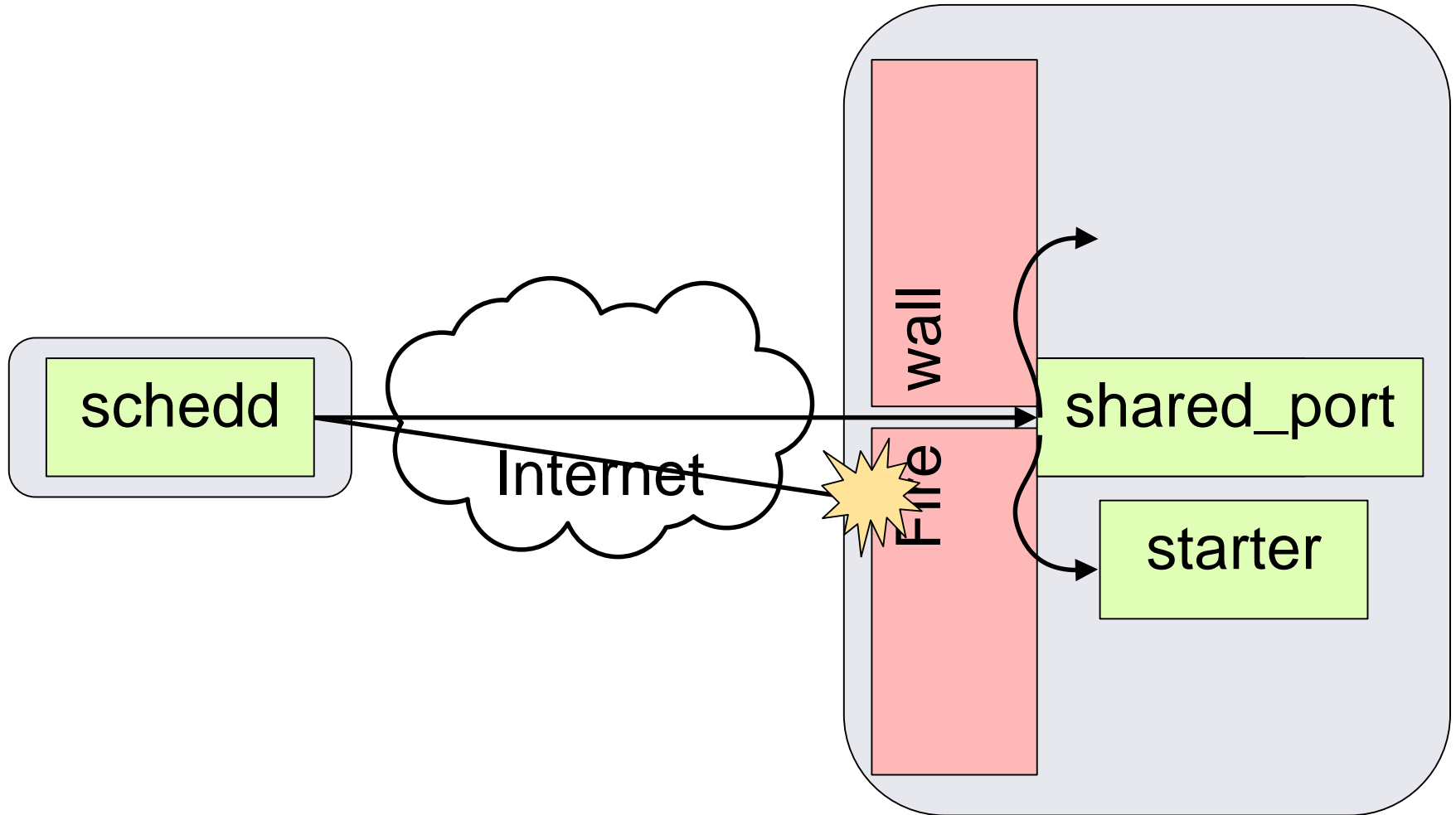


# Shared Port

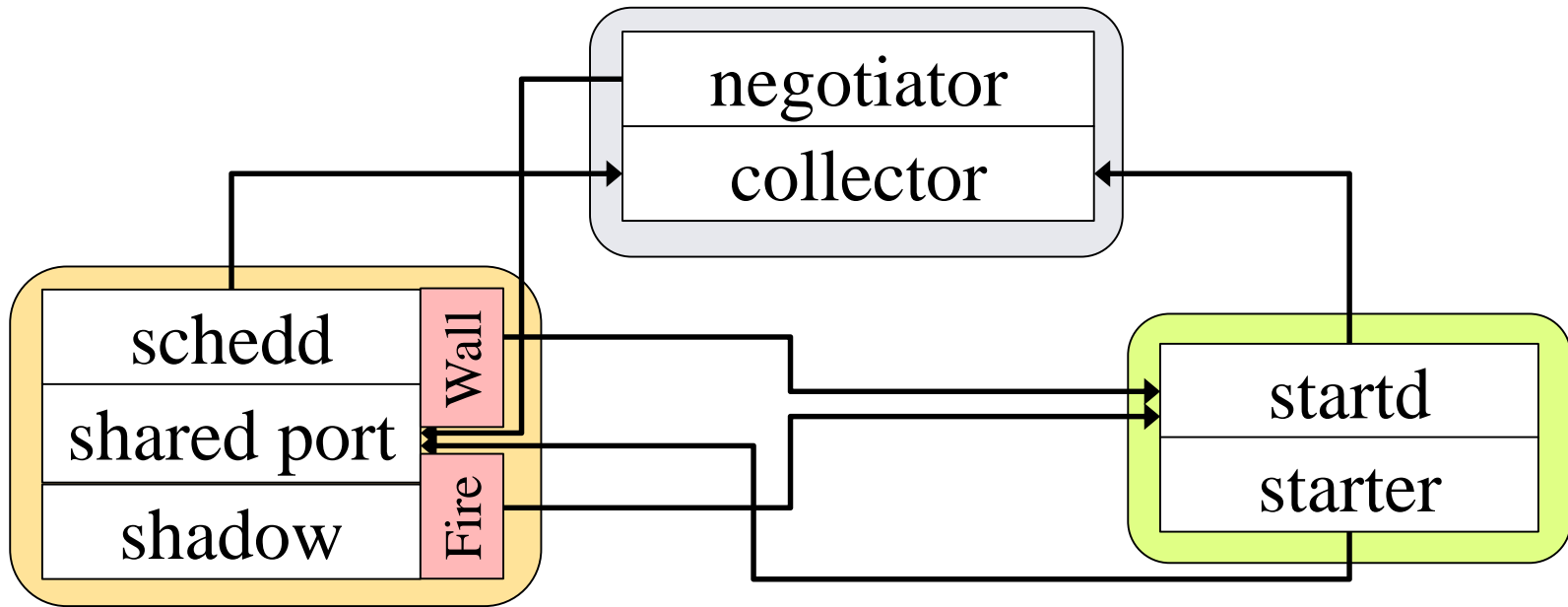
- › Problem: Firewall
  - Admin willing to open only one port
- › Problem: only ~60k TCP ports
  - Need one per shadow
- › Shared Port Service
  - Listens on single port for incoming connections
  - Hands each connection to intended recipient



# Shared Port



# Firewalled Submit Node



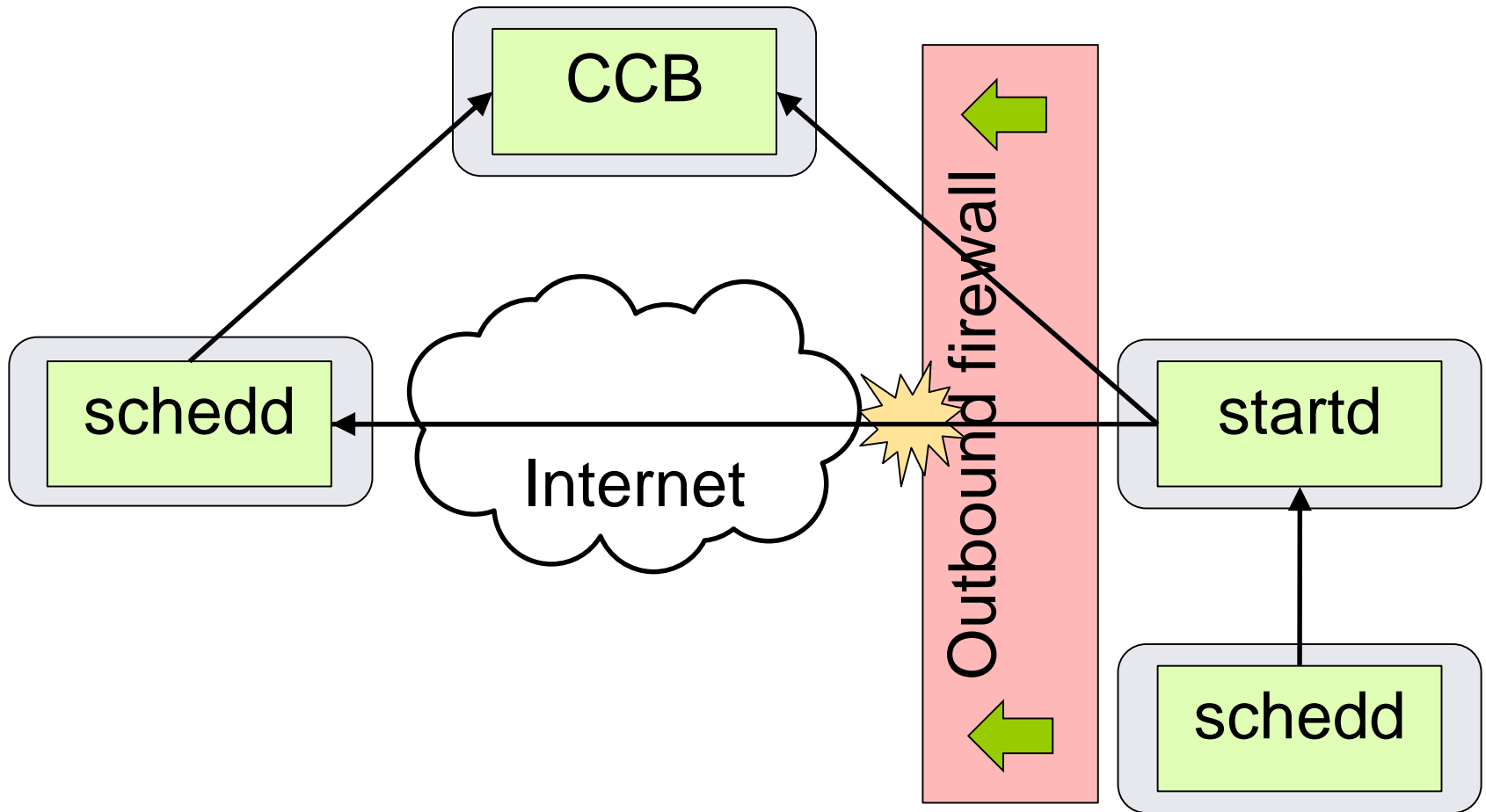
# TCP Forwarding Host

- › Problem: Private network with NAT
- › Traverse firewall via port forwarding
  - Allocate a public IP address
  - Connections to public address forwarded by NAT to machine on private network
- › Common in the Cloud

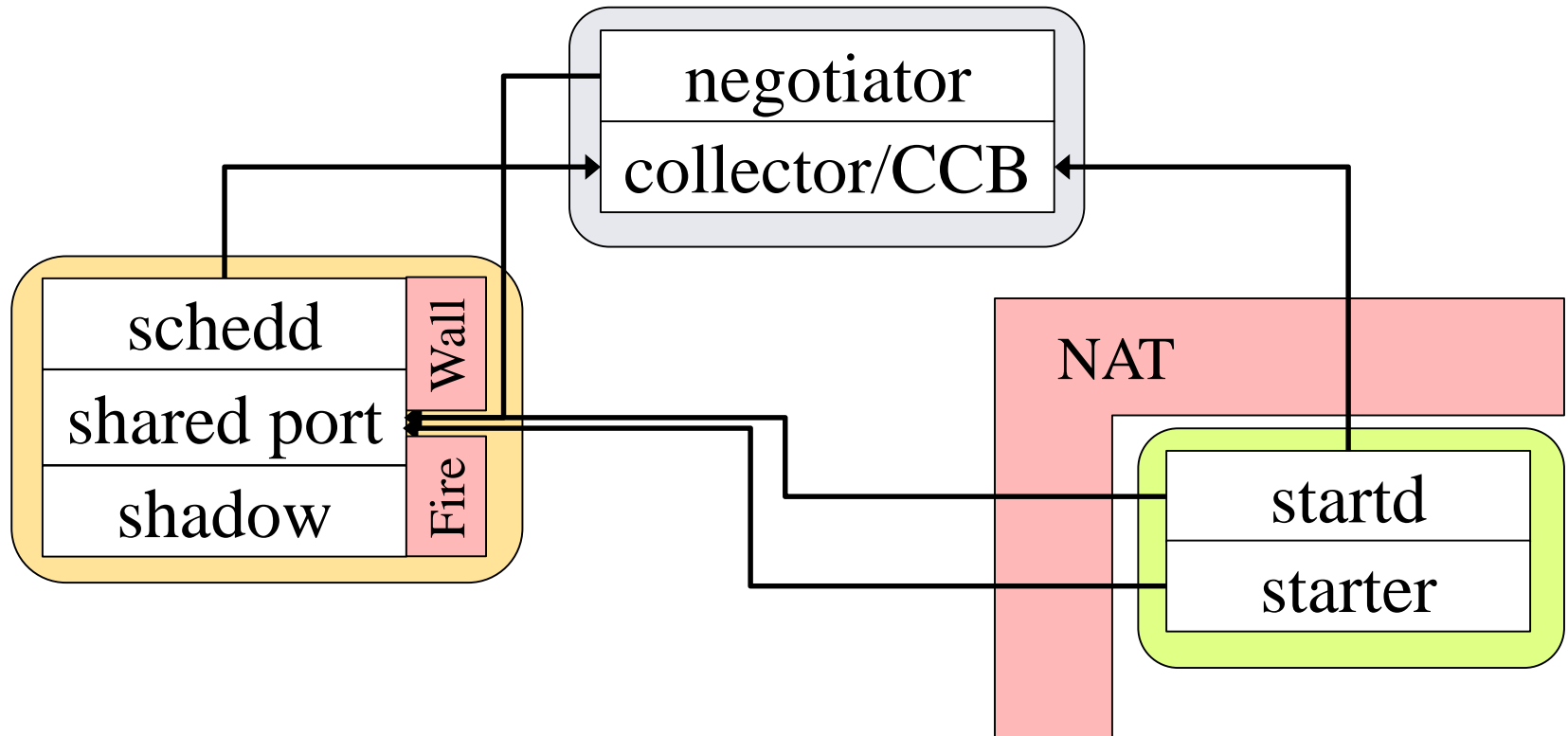
# Condor Connection Broker

- › Problem: Private network with NAT
  - Or firewall with no opening for HTCondor
- › Traverse firewall by reversing connection
  - Client sends connection request via broker
  - Server initiates TCP connection to client
- › Only bypasses one firewall
  - Client and broker (CCB server) must have publically routable addresses

# CCB: Condor Connection Broker



# NATd Execute Nodes



# Port Usage (Digression)

- › Shadow for each running job
- › In fairy-tale setup
  - Each shadow uses two ports
  - Limit of ~25k running jobs
- › With shared port and CCB
  - Shadow use no ports
  - No network limit on number of running jobs

# Quiz

1. Why do schedds and central managers need to be mixed-mode in a pool split between IPv4 and IPv6 nodes?
2. Why use CCB on execute nodes?
3. Why use both CCB and shared port?
4. If both the schedd and the execute nodes are NATd, what do you do?



# Question 1

- › Why do schedds and central managers need to be mixed-mode in a pool split between IPv4 and IPv6 nodes?
  - They need to be able to talk to all execute nodes

# Question 2

- › Why use CCB on execute nodes (and not submit nodes)?
  - Easier to make submit nodes publically accessible (fewer of them)

# Question 3

- › Why use both CCB and shared port?
  - Can't use CCB for both schedd and startd
  - No ports used for shadow, so no limit on number of running jobs

# Question 4

- › If both the schedd and the execute nodes are NATd, what do you do?
  - If same NAT, no problem
  - TCP Forwarding Host for schedd

# Congratulations!

