Announcements

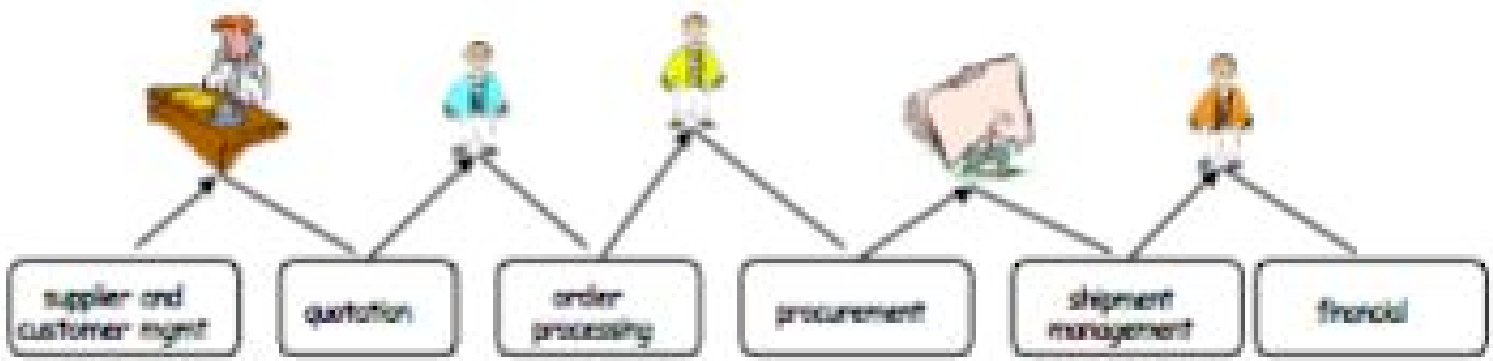
- Project proposal assignment is open, due tonight by midnight
- See instructions for CMS group formation
  - *do this first*

EAI - Workflow - Web Technologies

- Message Brokers, EAI, Workflow
  - [ACKM04] Ch 3
- Web Technologies, J2EE Introduction
  - [ACKM04] Ch 4

Business Process

- A Supply Chain ...



- This is really a *sequence* of atomic transactions
- Some txns could happen concurrently ...

Business Process - Workflow

- Why do things this way on computer?
- Lots of reasons ...
  - Availability
    - need all the resources at once
  - Resource contention
    - hold only the subset of resources you needed *now* => better throughput
  - Legacy systems
    - may not have distributed commit

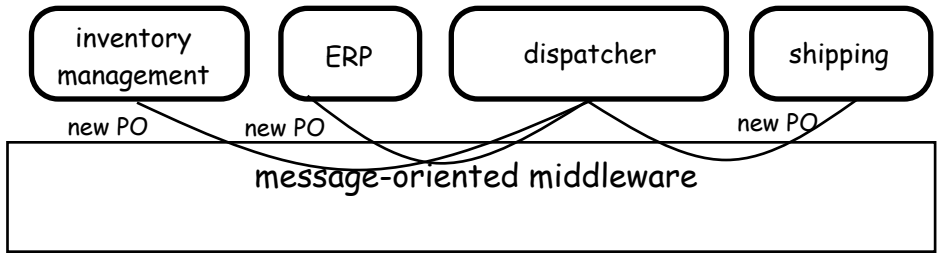
ACID Properties?

- Clearly fails to provide Isolation
  - other txns can read state between committed steps
  - e.g. transfer funds by withdraw and deposit as separate txns
  - can view this as expanding set of "consistent" states
  - highly application dependent

ACID Properties?

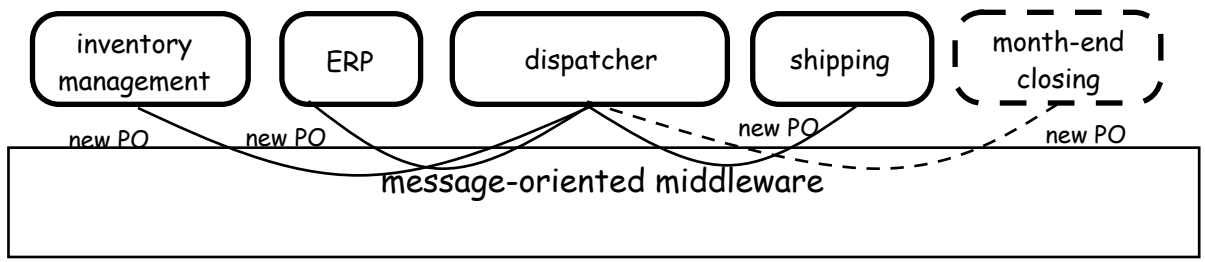
- Atomicity?
  - not in short term -- same argument as for isolation!
  - long term -- to roll back a sequence of transactions, execute a *compensating transaction* for each committed step
  - but this is not always possible
    - S1 makes deposit to bank acct
    - S2 makes withdrawal
    - Later we try to compensate for S1's deposit and find insufficient funds ...

Enterprise Application Integration (EAI)



- Inventory, ERP and Shipping are legacy apps
  - all run independent transactions
- New dispatcher app integrates them

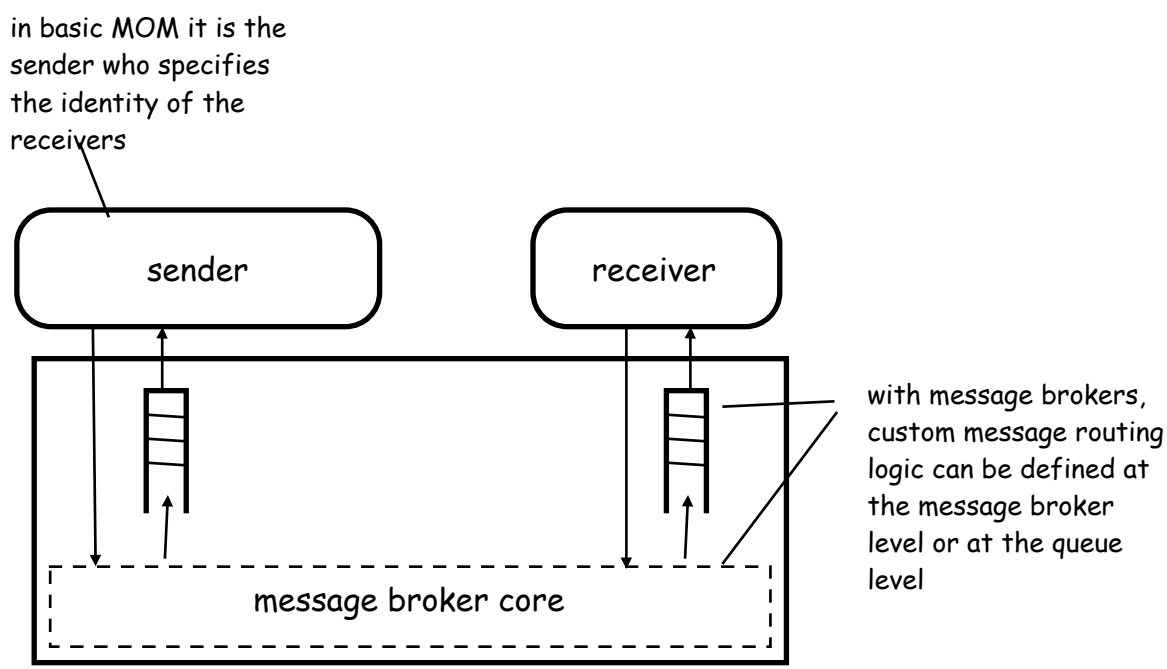
EAI



- With traditional MOM, integrating another application requires changing the dispatcher
- Is this really necessary?

Message Broker

- Message broker determines destinations
  - based on sender identity, message type, message content
- Senders do not specify or know who the receivers are!



Publish / Subscribe

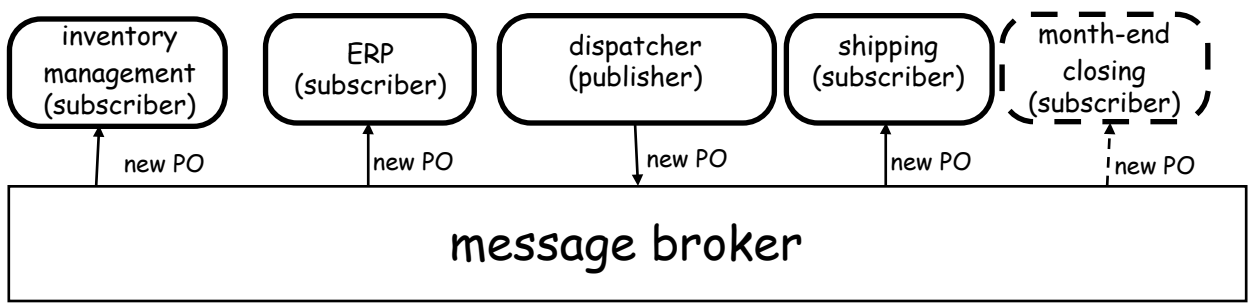
- Publication:
  - enqueue a message
- Subscription: specifies
  - message type (e.g. purchase order)
  - Boolean filter function on messages
- Message broker sends copy of each publication to each matching subscription
- Persistent?

Variant: Java Message Service

- Explicit *topic* takes place of message type
- Publication:
  - Publisher specifies a topic
- Subscription:
  - Subscriber specifies topic and filter predicate

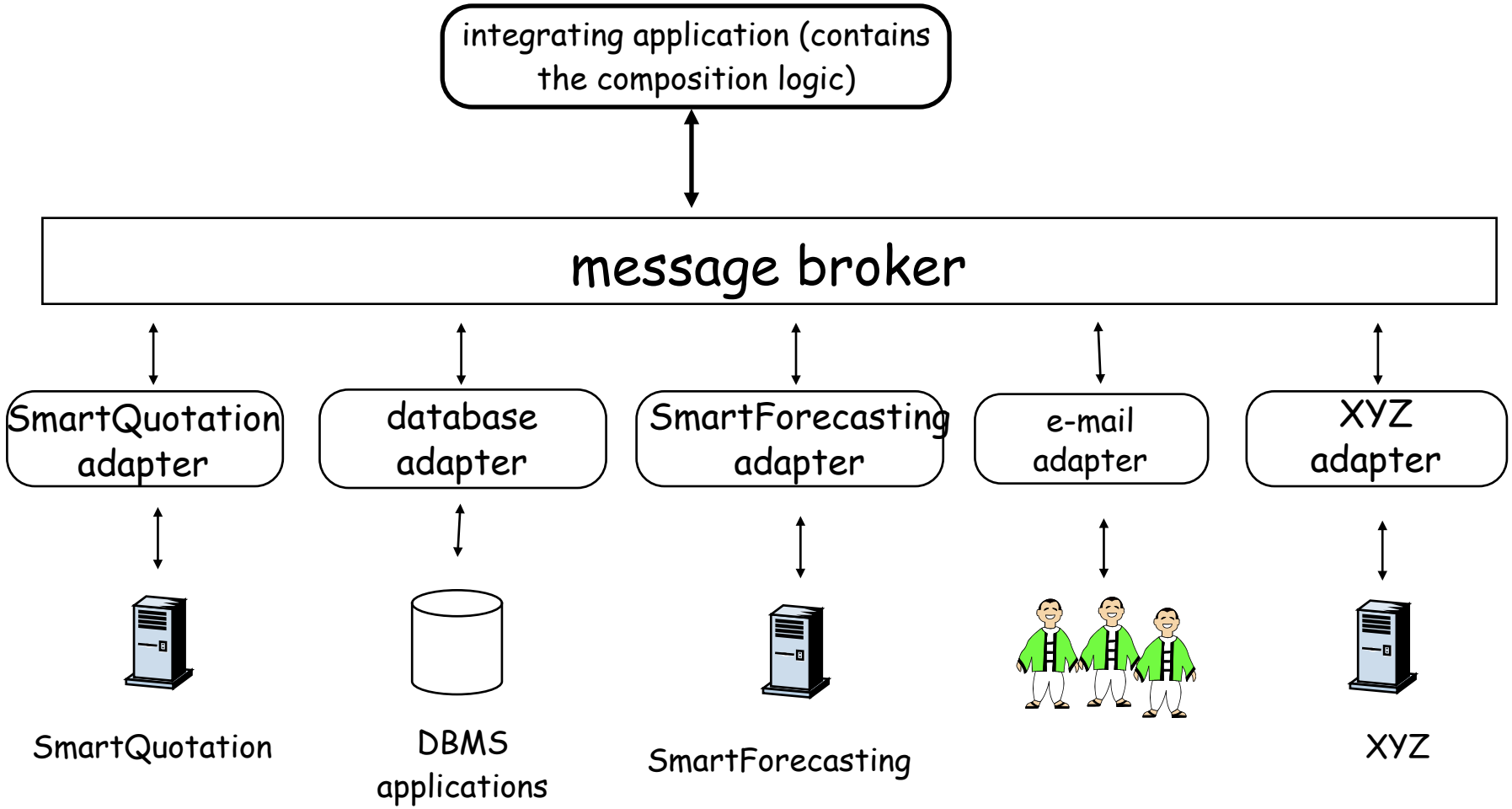
EAI Using Message Broker

- Add new application by altering routing specification in Message Broker ...
- Publish / Subscibe
- Fine as long as new application expects existing message type



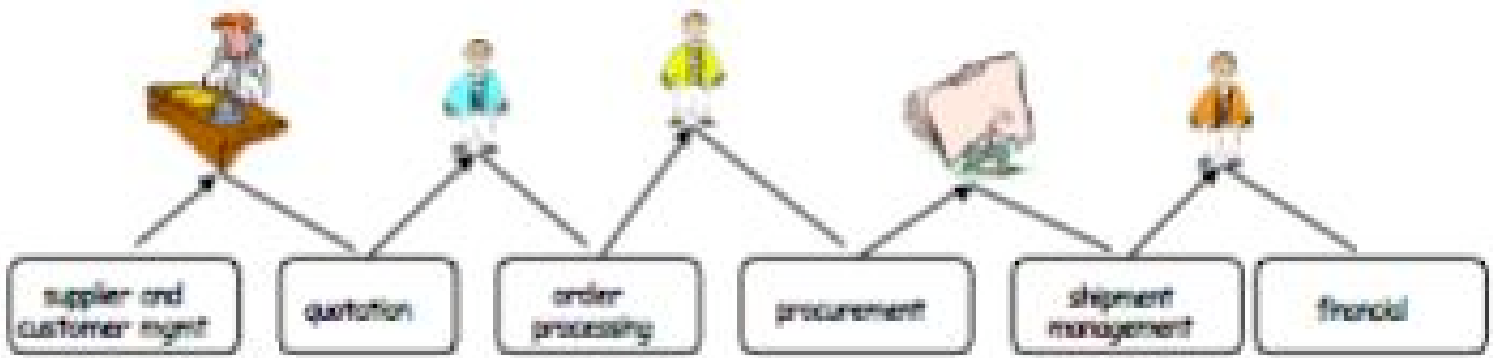
More Precisely ...

- Dispatcher is new integrating application
- Legacy applications get adapters



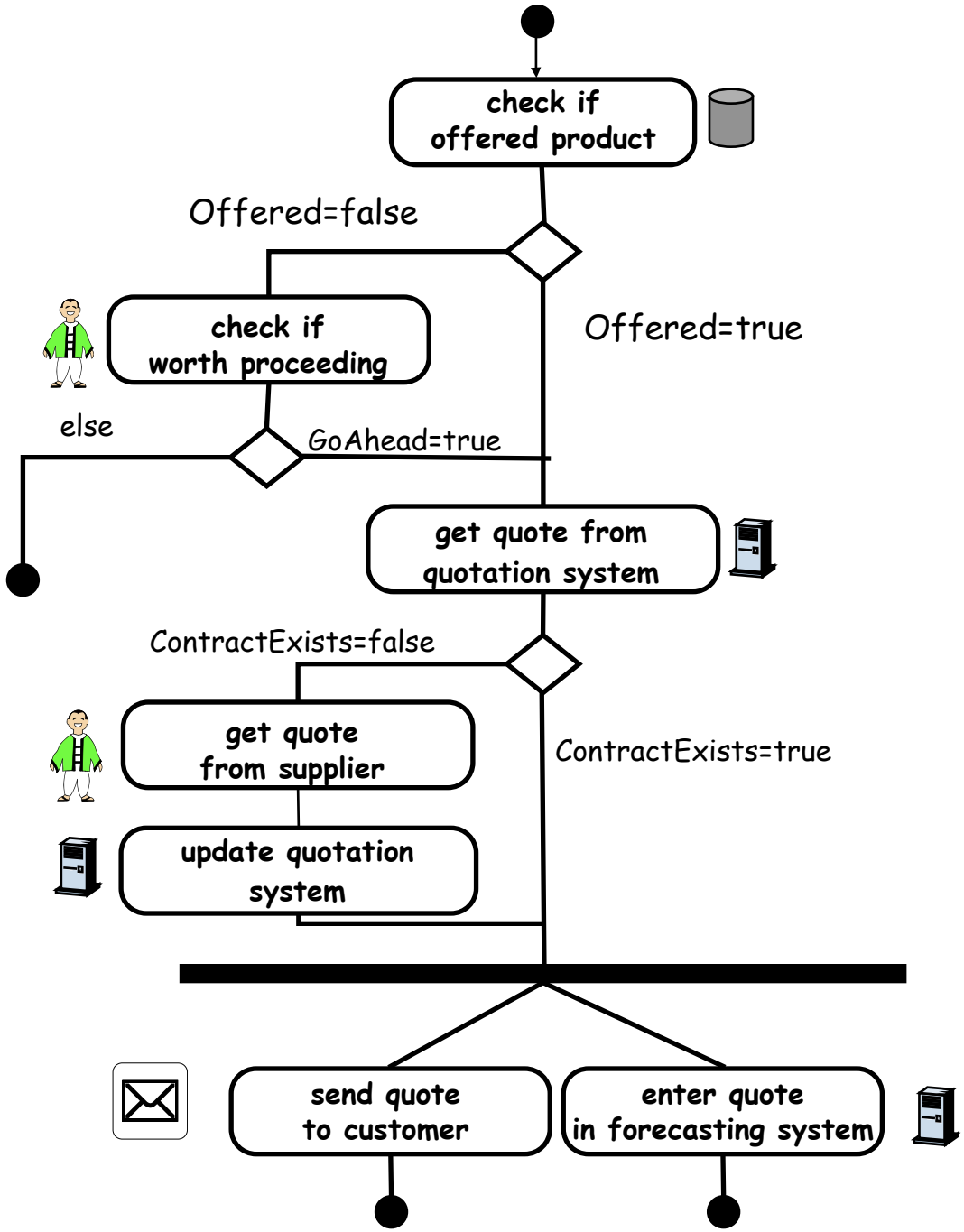
Recall: Business Process

- A Supply Chain of automated and manual steps



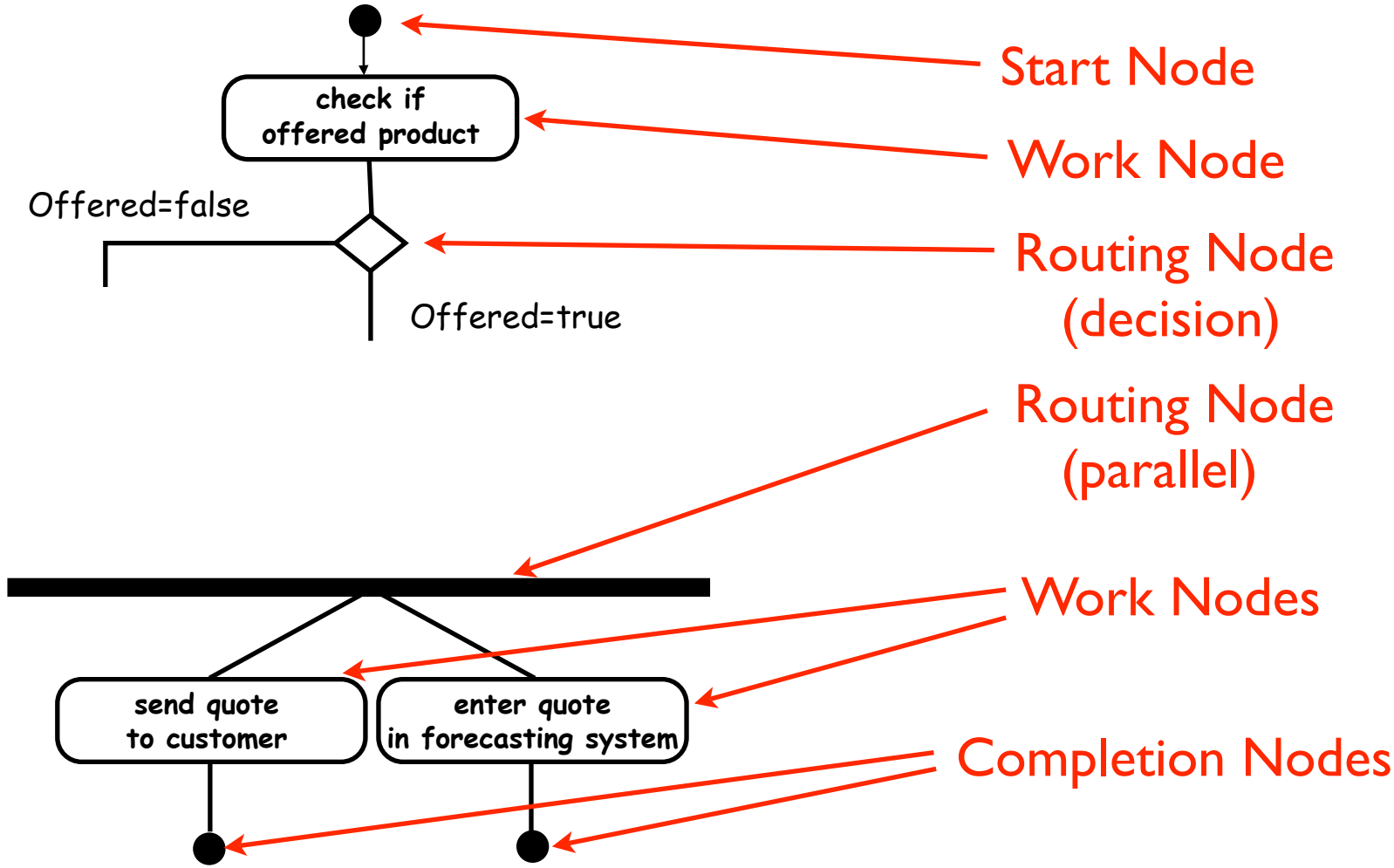
- WfMS supervises all steps
- Even the manual ones (using eMail)

Workflow Specification

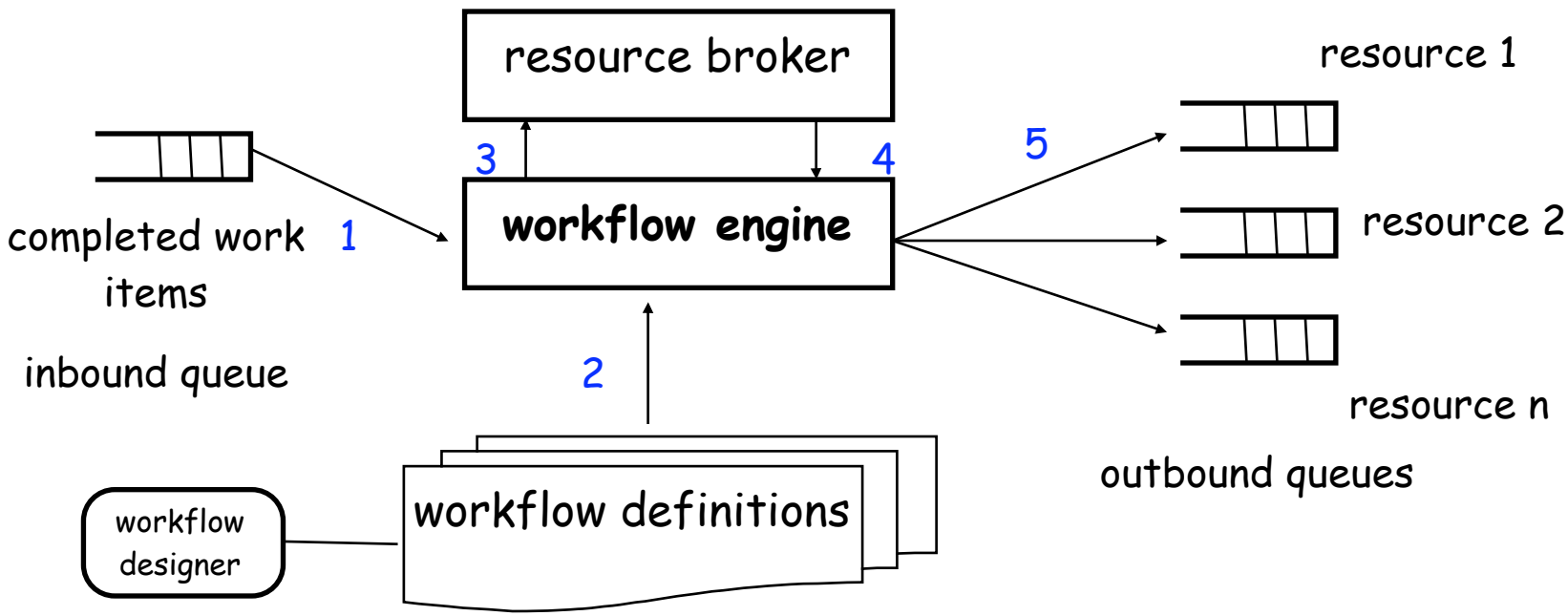


variables:
QuoteReferenceNumber: int
Customer: String
Item: String
Quantity: int
RequestedDeliveryDate: Date
DeliveryAddress: String
GoAhead: Bool
ContractExists: Bool
Offered: Bool

Workflow Specification



Workflow System



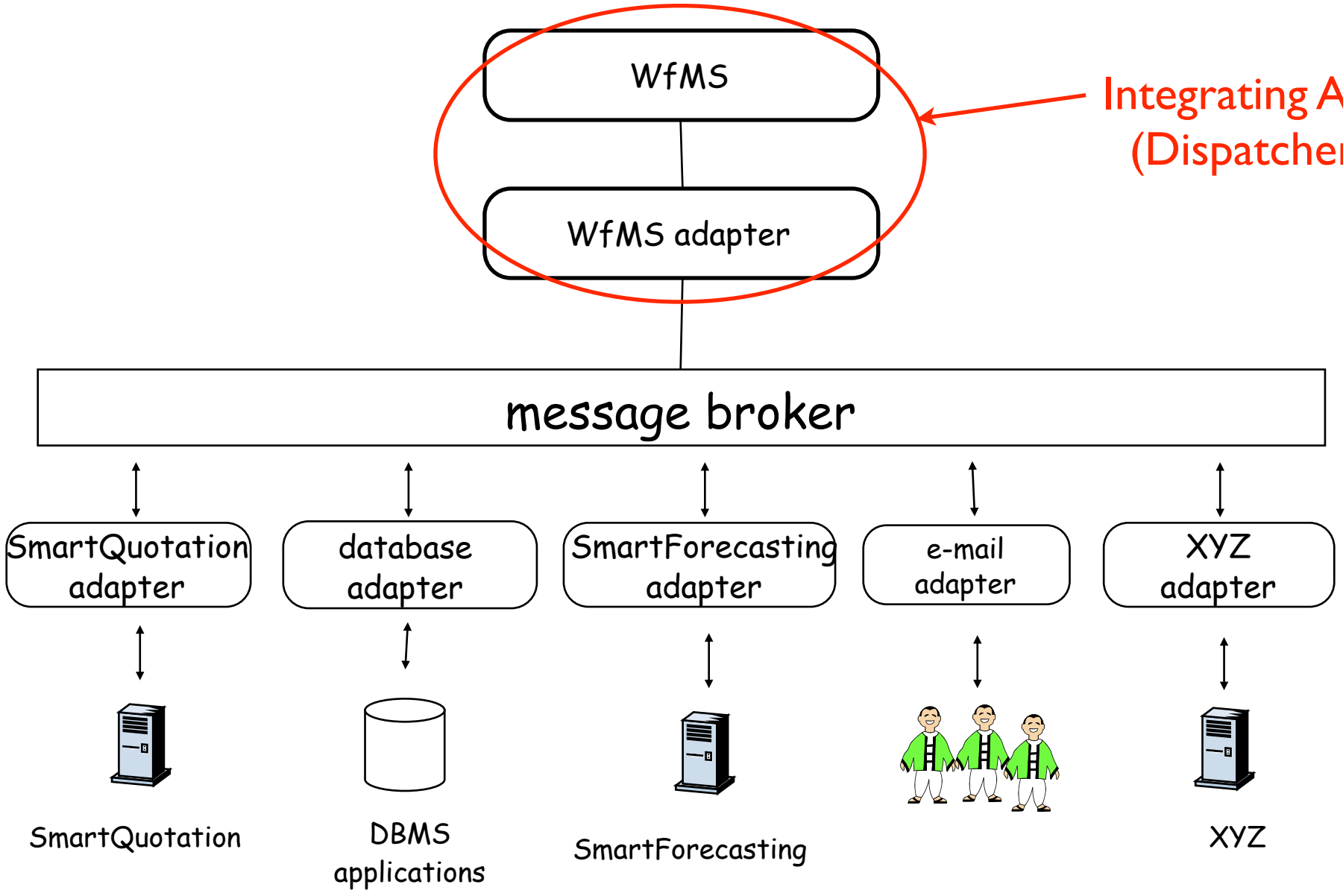
Error Conditions

- Deadline for step completion
  - WfMS takes corrective action
- Exception detected in application
  - Several mechanisms to raise exception

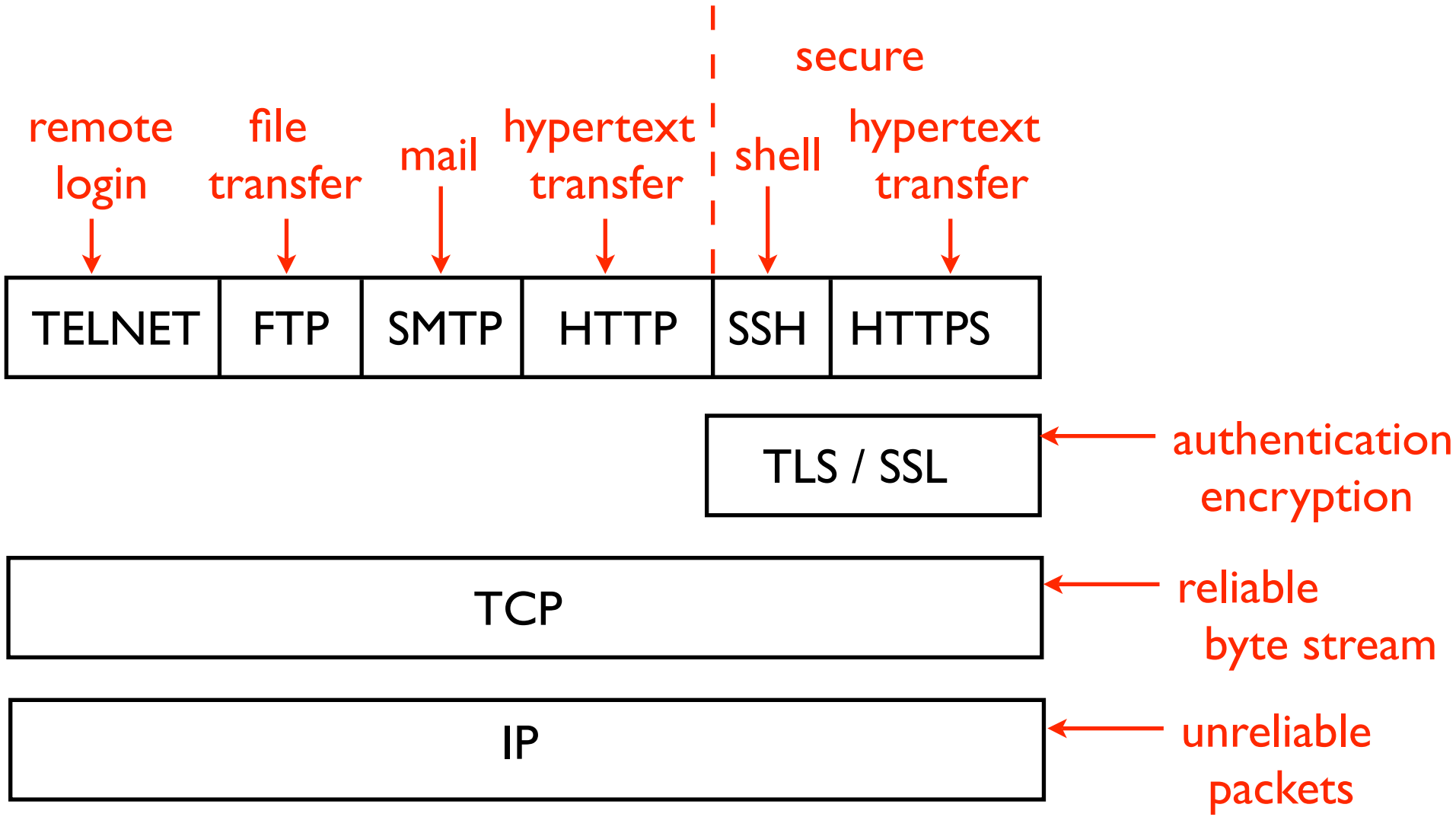
Error Conditions - Recovery

- Forward Recovery
  - State of workflow maintained by WfMS
  - Server fails => restart from that point
  - Like recovery from transactional RPC
- Backward Recovery
  - Execute compensating action to undo each previous action of workflow *in reverse order*
  - Like rollback in Sagas

Combining - Integrating App is WfMS



Internet Protocols



HTTP is really simple

Request (GET/PUT/POST/OPTIONS/DELETE)
URI
Message

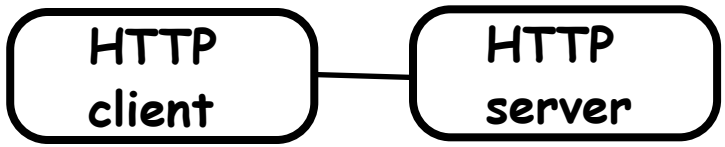
Status
Message

.
.
.

HTTP is really simple

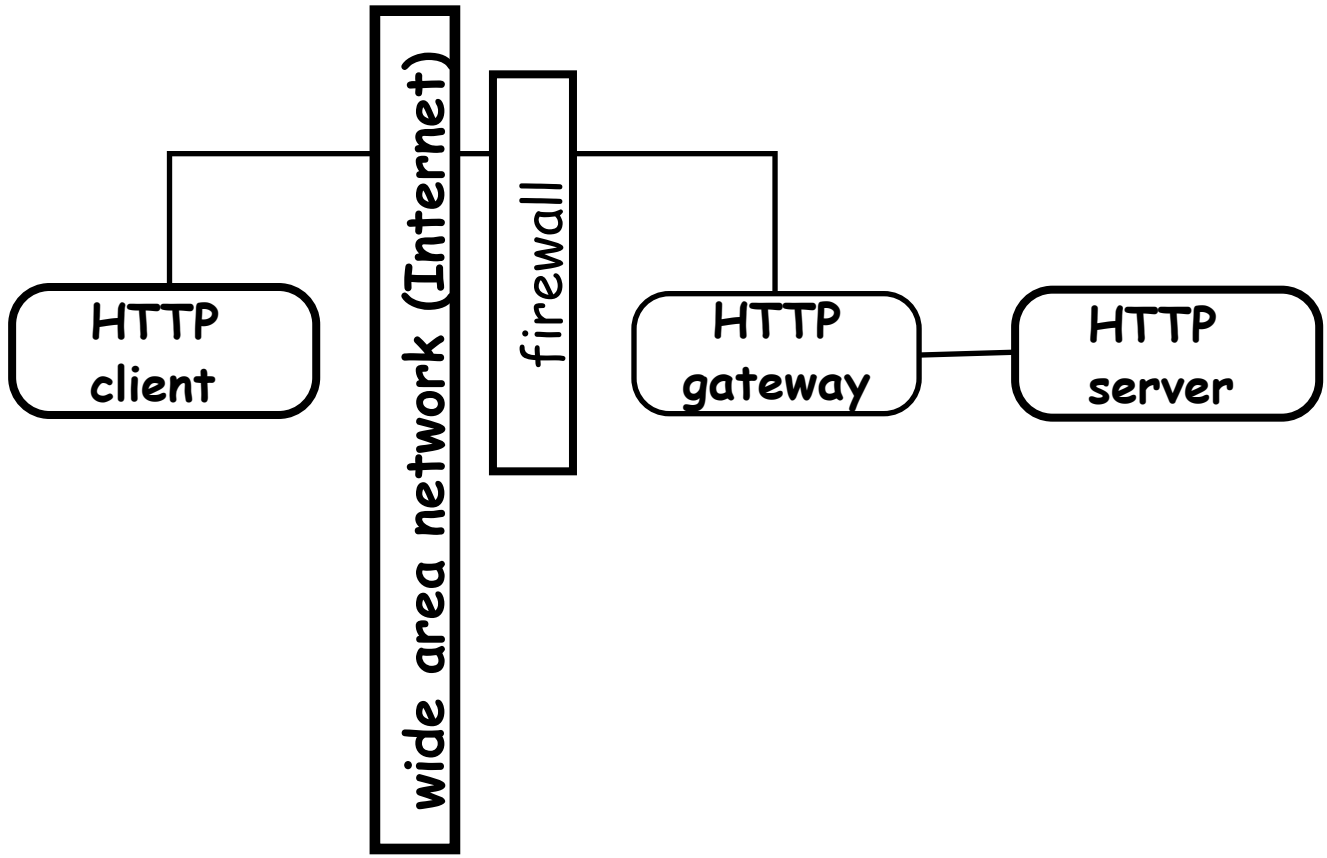
- URI: think URL (host + path)
  - server.dnsname.com/file/path/name
- Message: think MIME encoded file
  - arbitrary sequence of bytes of identified type
- Persistent connection

Simple HTTP Connection



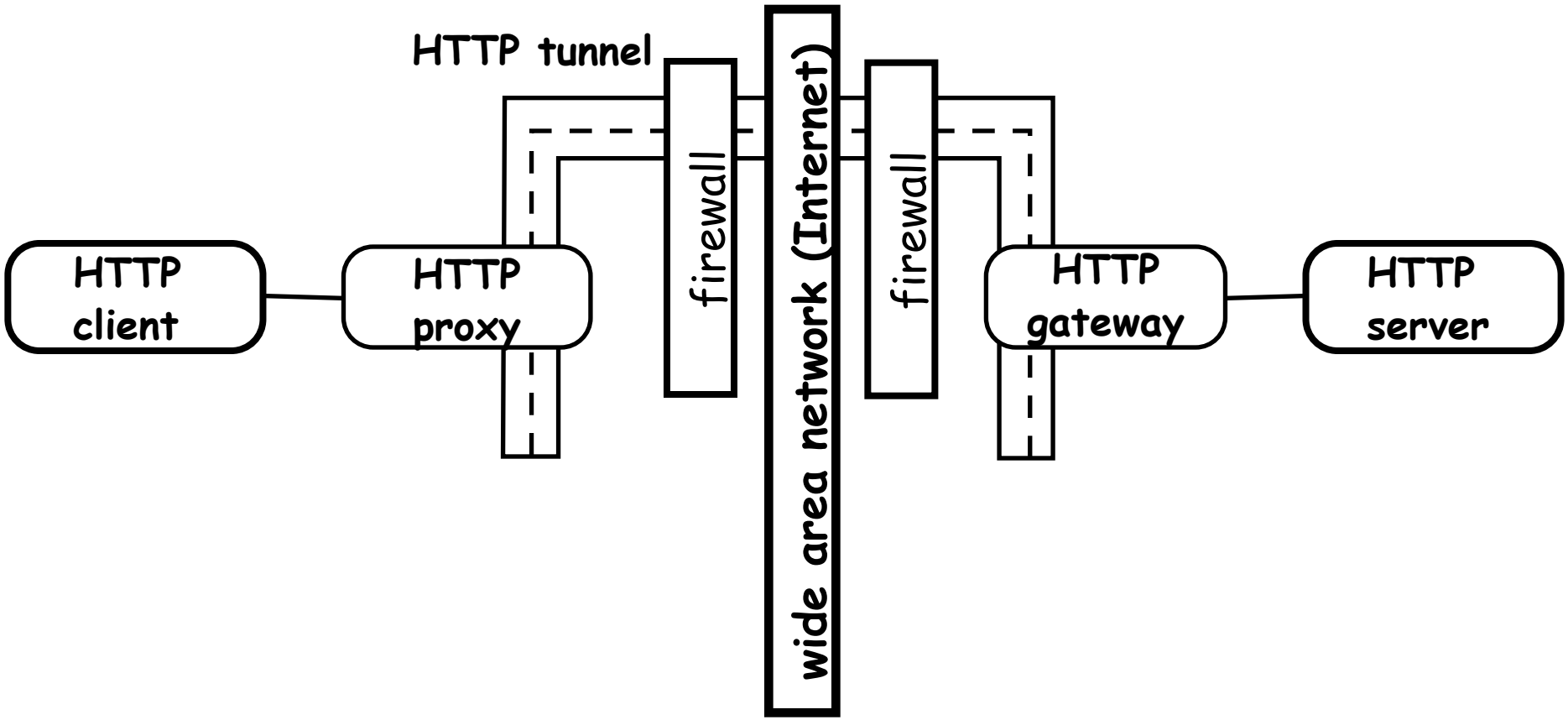
- Occasionally
  - if client and server are both behind the same firewall
  - And they trust one another!

HTTP Connection



- Server behind firewall and HTTP gateway
- More realistic

HTTP Connection

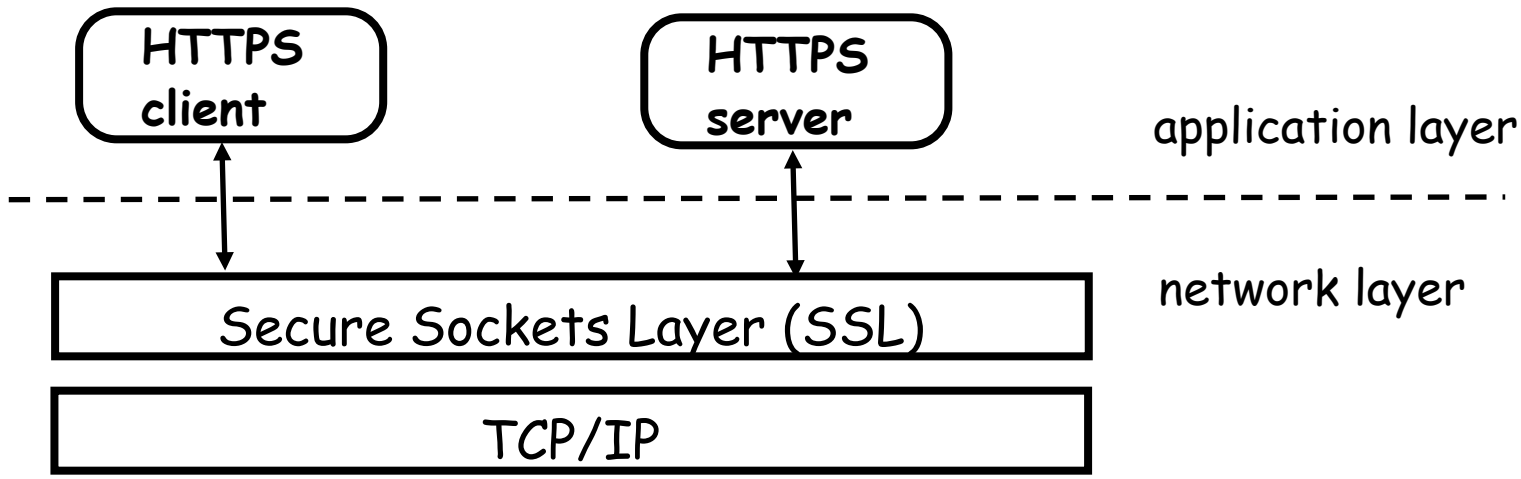


- Two firewalls, HTTP proxy, HTTP gateway

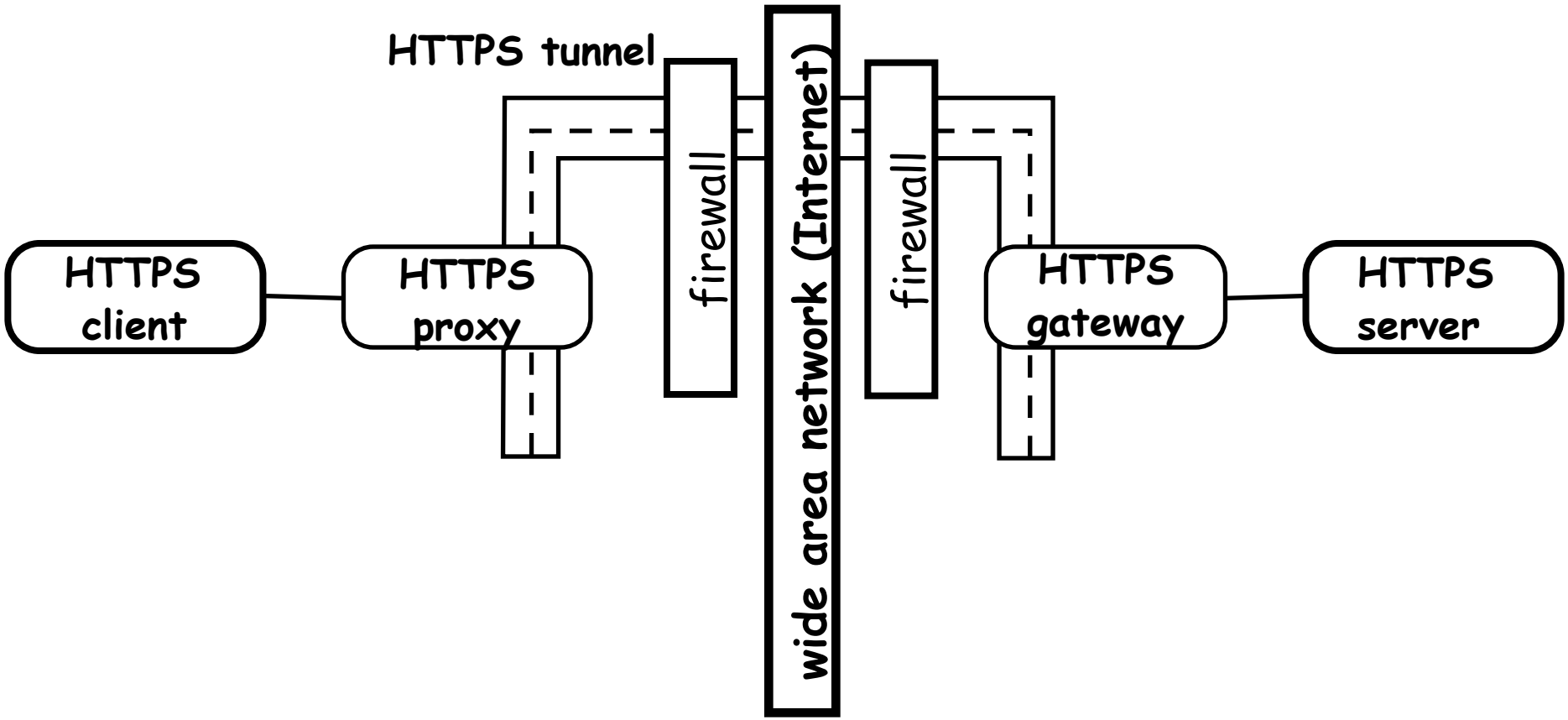
SSL

- Secure (encrypted) channel between client and server
- Server authenticated to client
- Client optionally authenticated to server
  - Public Key certificates

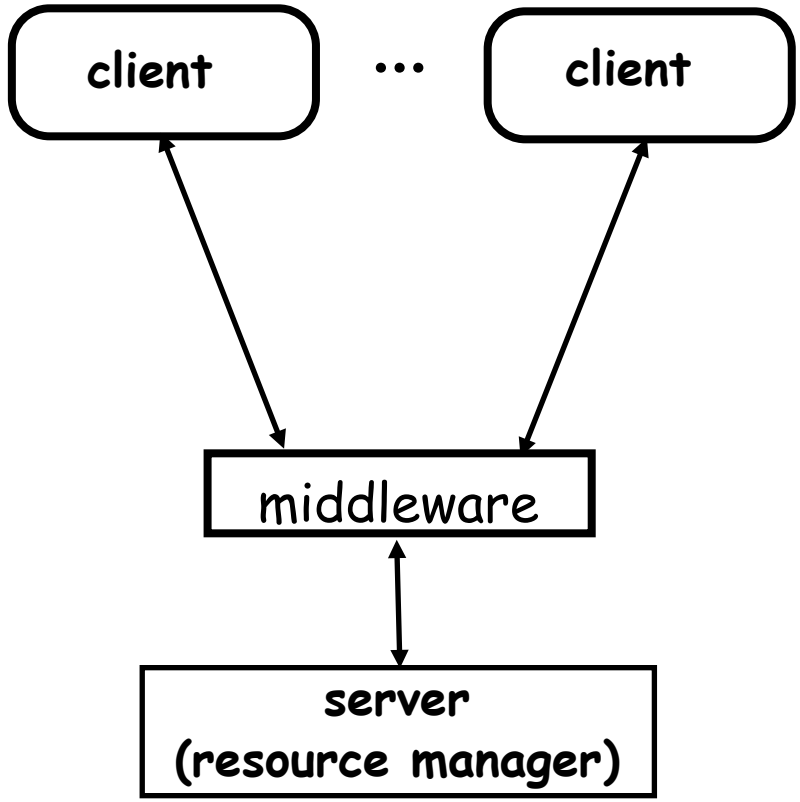
HTTPS - SSL


HTTPS client
HTTPS server
application layer
network layer
Secure Sockets Layer (SSL)
TCP/IP

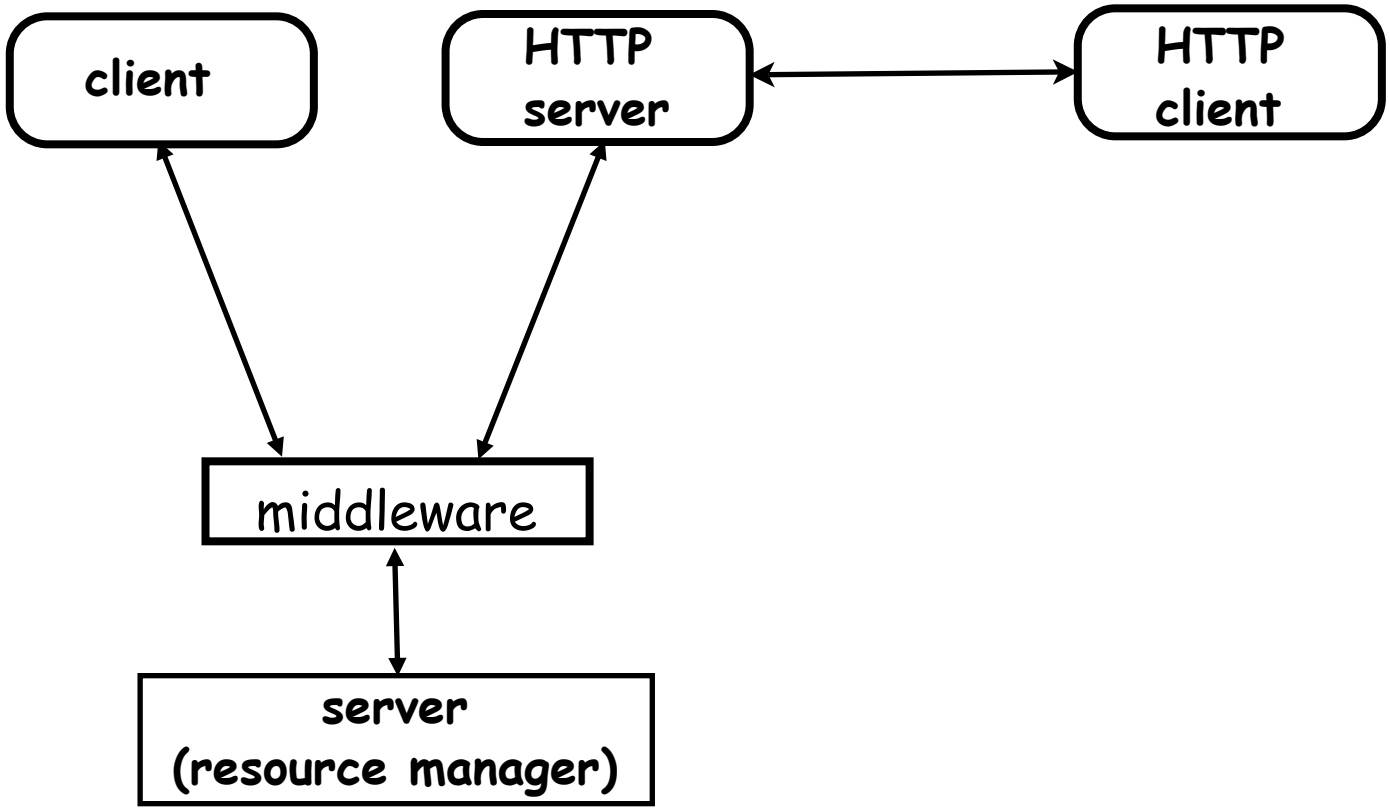
HTTPS Connection



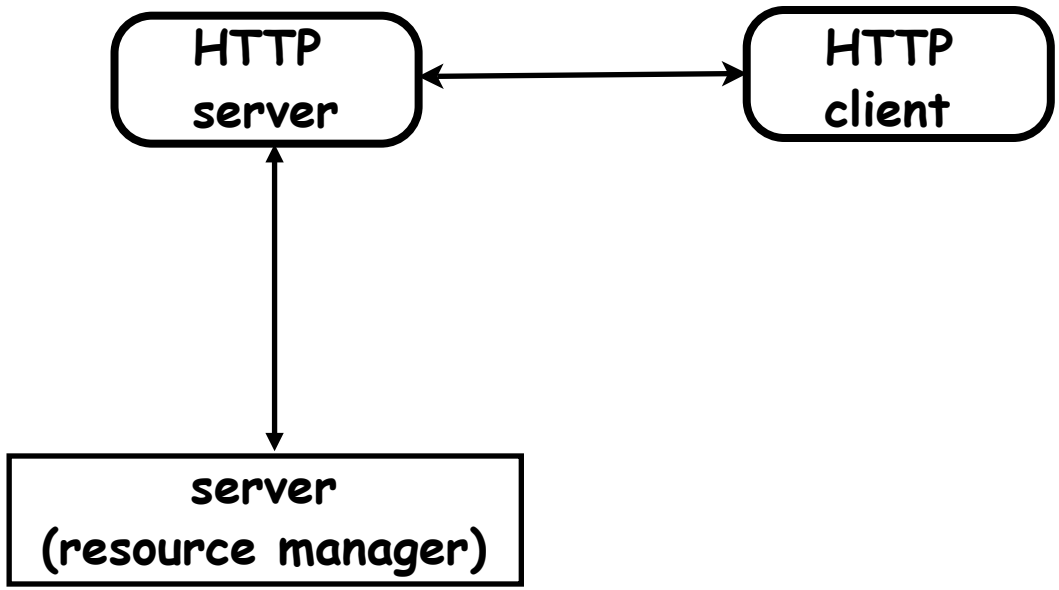
Recall 3-Tier



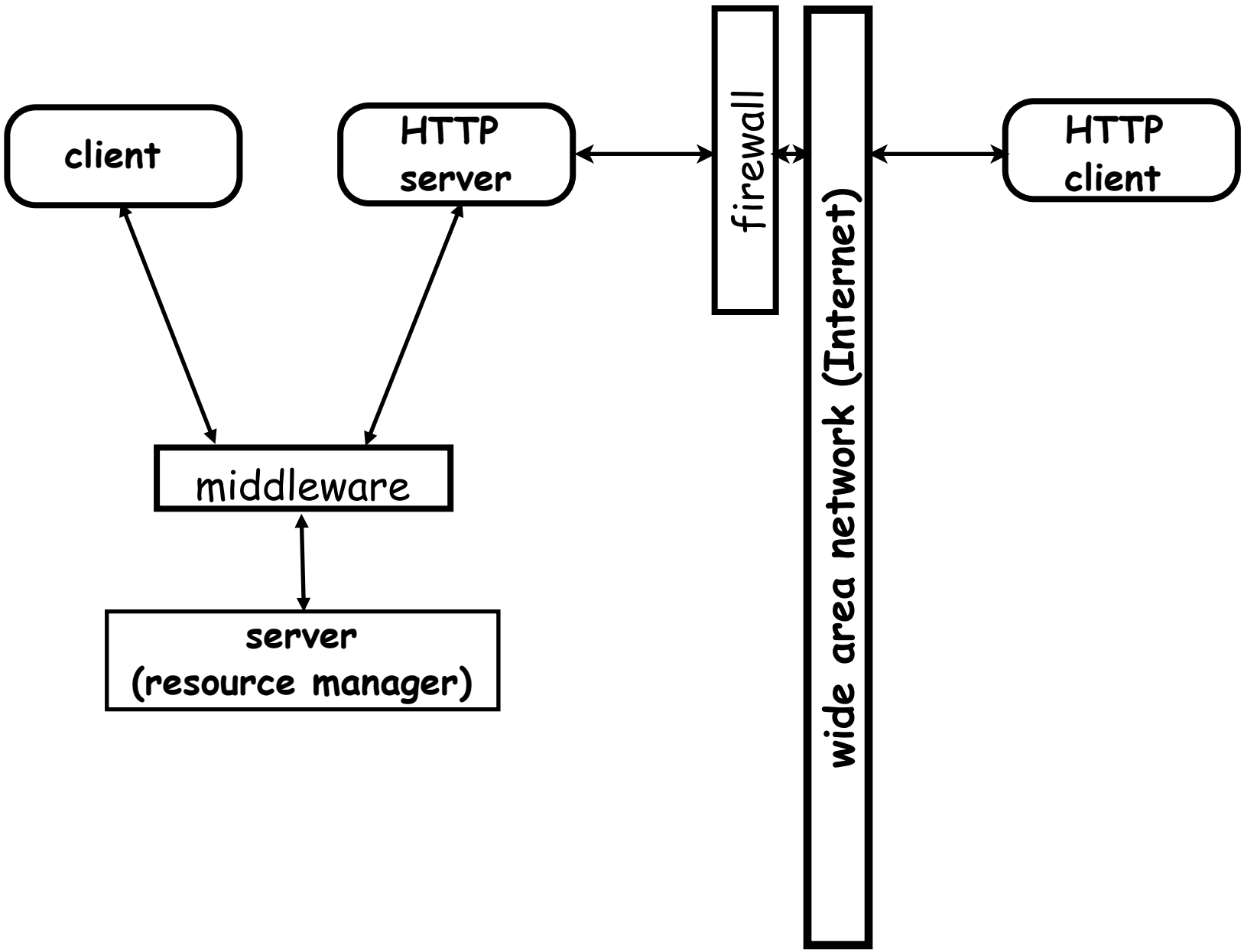
Adapter to HTTP client



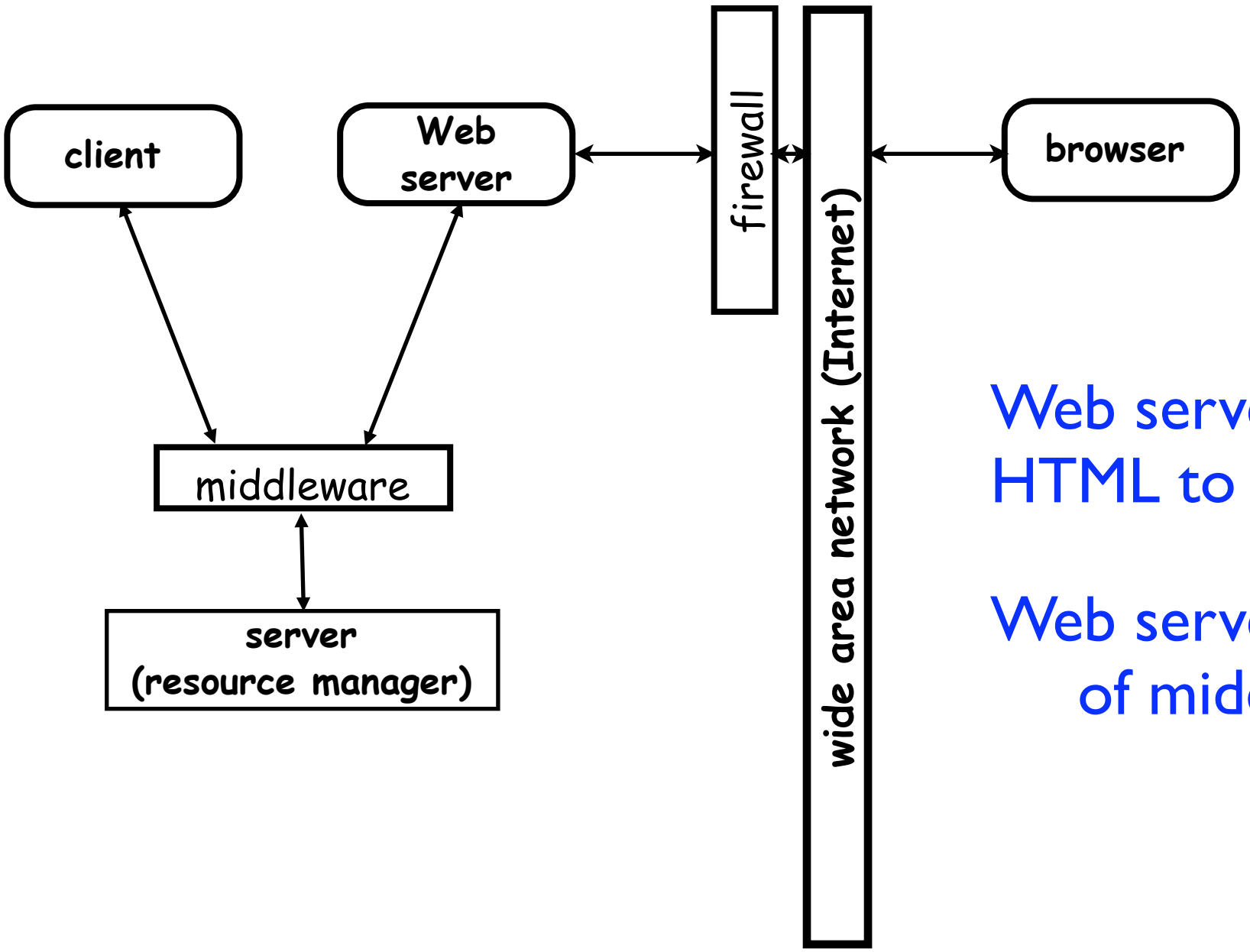
This also works 2-Tier



Over the Web


client
HTTP server
firewall
wide area network (Internet)
HTTP client
middleware
server (resource manager)

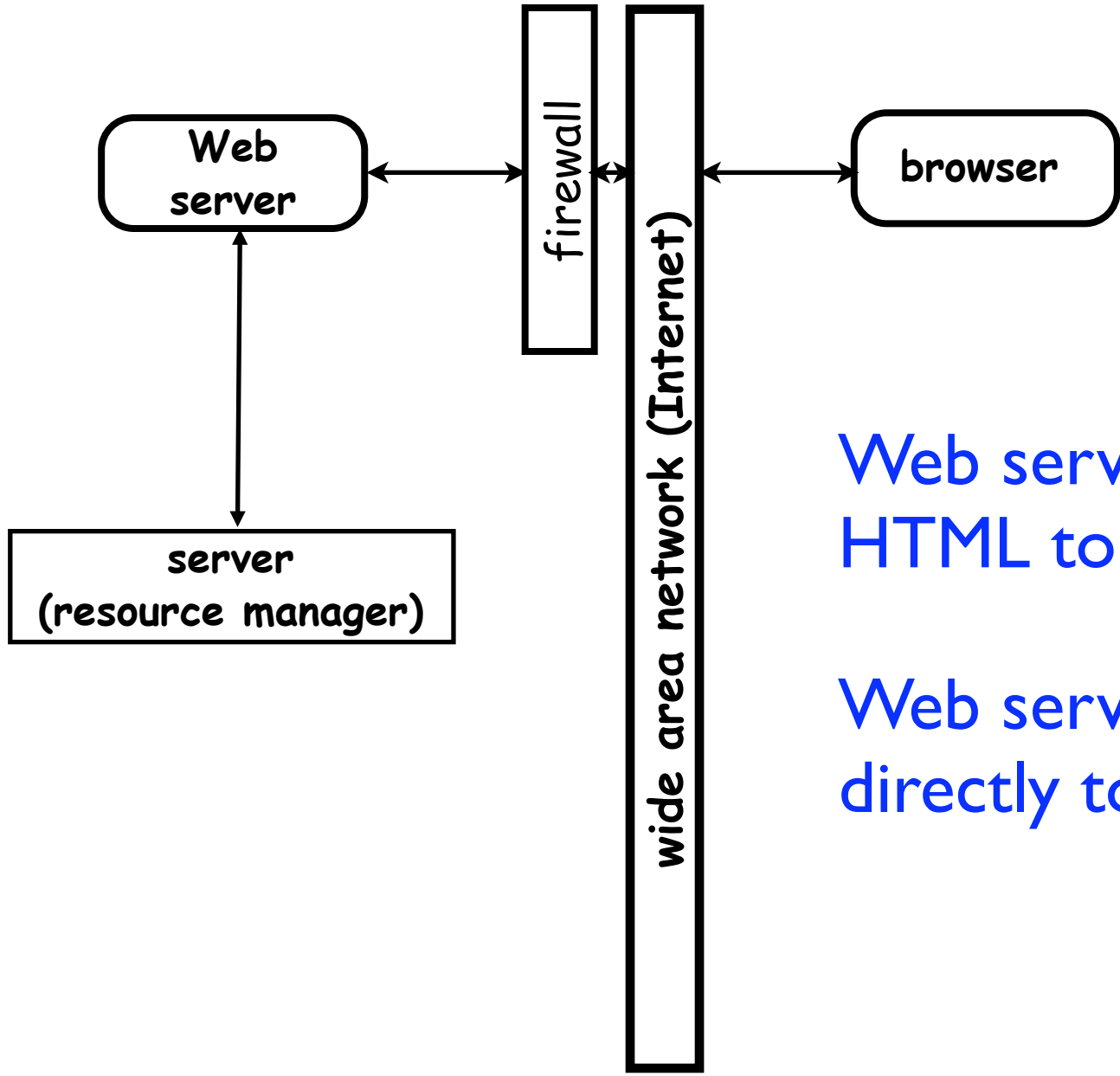
Client is a browser



Web server sends HTML to browser

Web server is client of middle tier

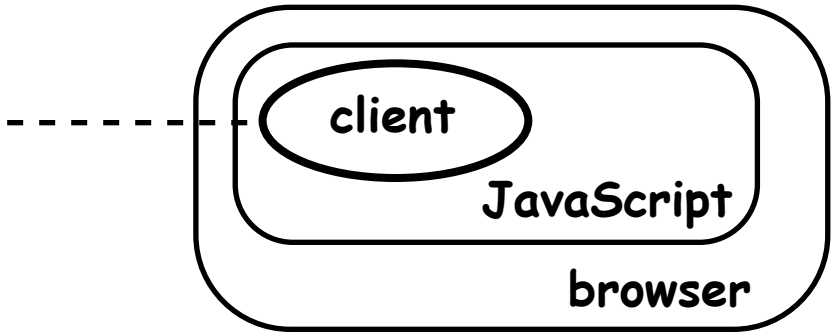
Still Works 2-Tier



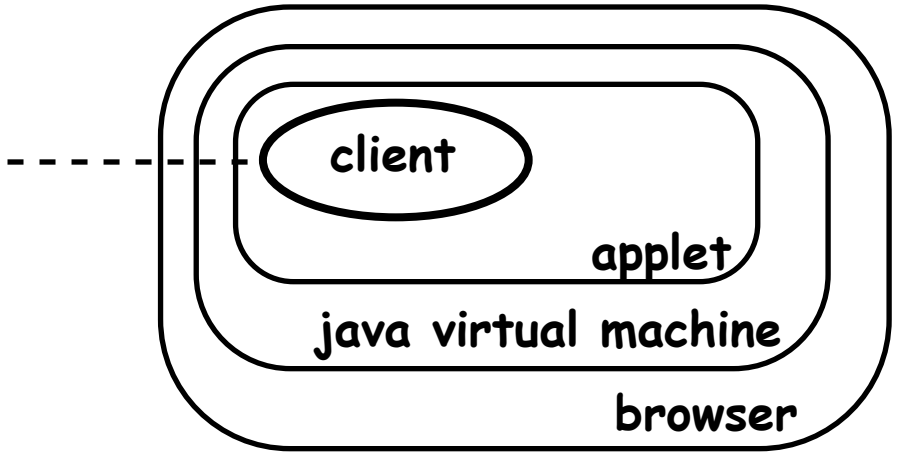
Web server sends HTML to browser

Web server talks directly to database

Thin Client in Browser



JavaScript
VBScript



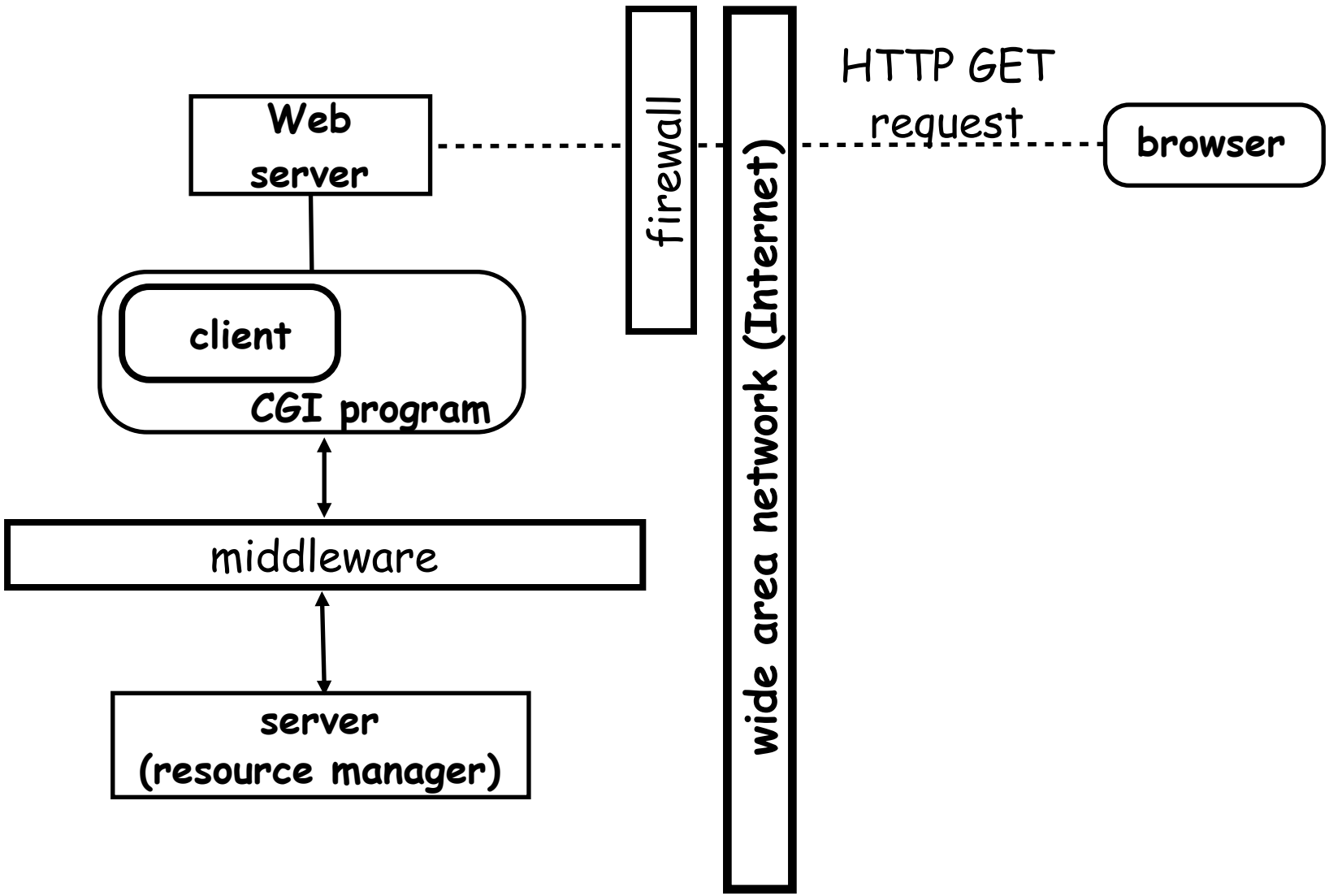
Applets

- HTML forms
- cookies

Web Server is Client of Middleware

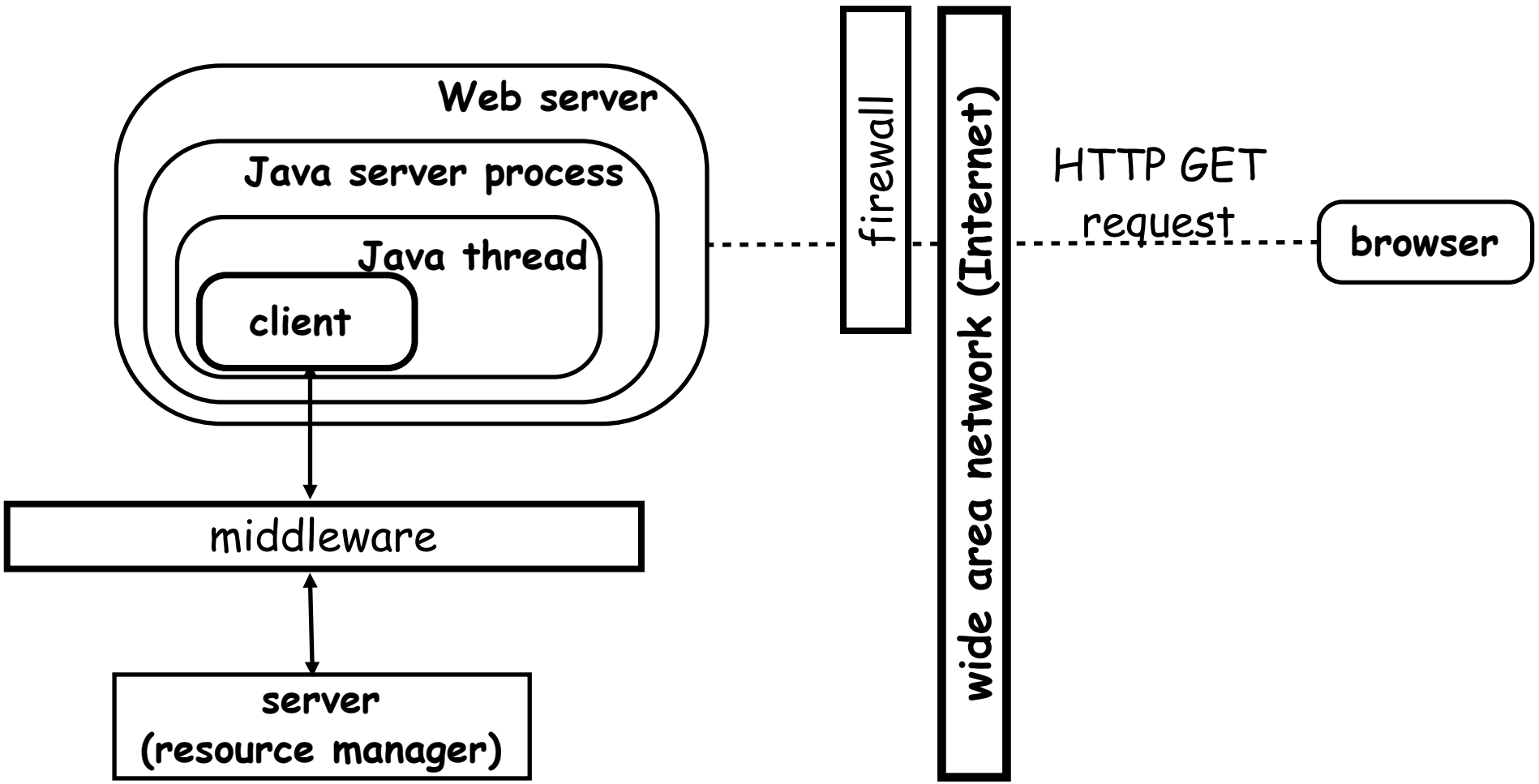
- Or maybe of the database!
- Technology to do this:
  - CGI
  - Servlets
- Ways to make web server execute code

CGI



- Create Process, run arbitrary program
- But very expensive

Servlets

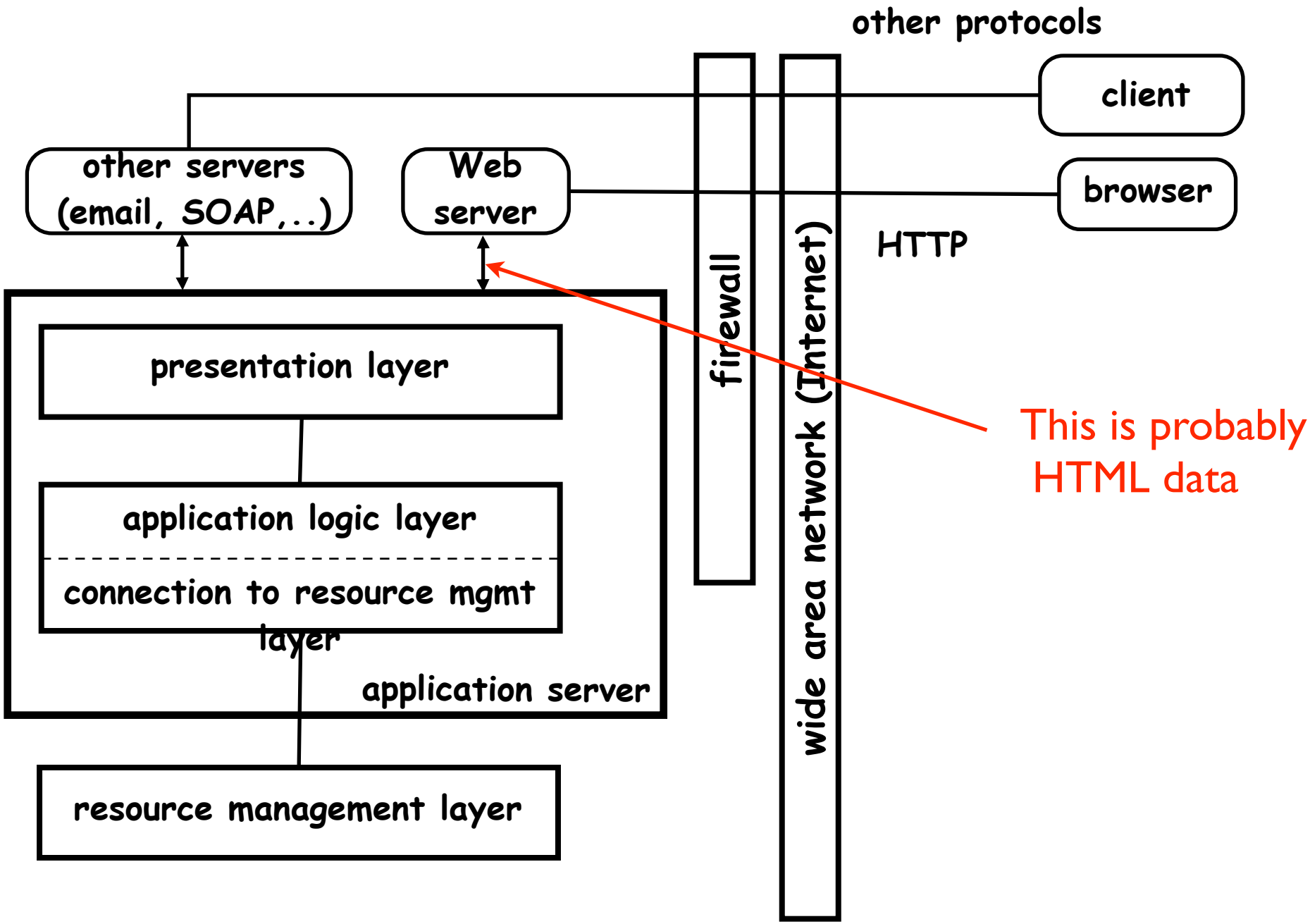


- JVM part of Web server
  - No process creation
  - Cache state

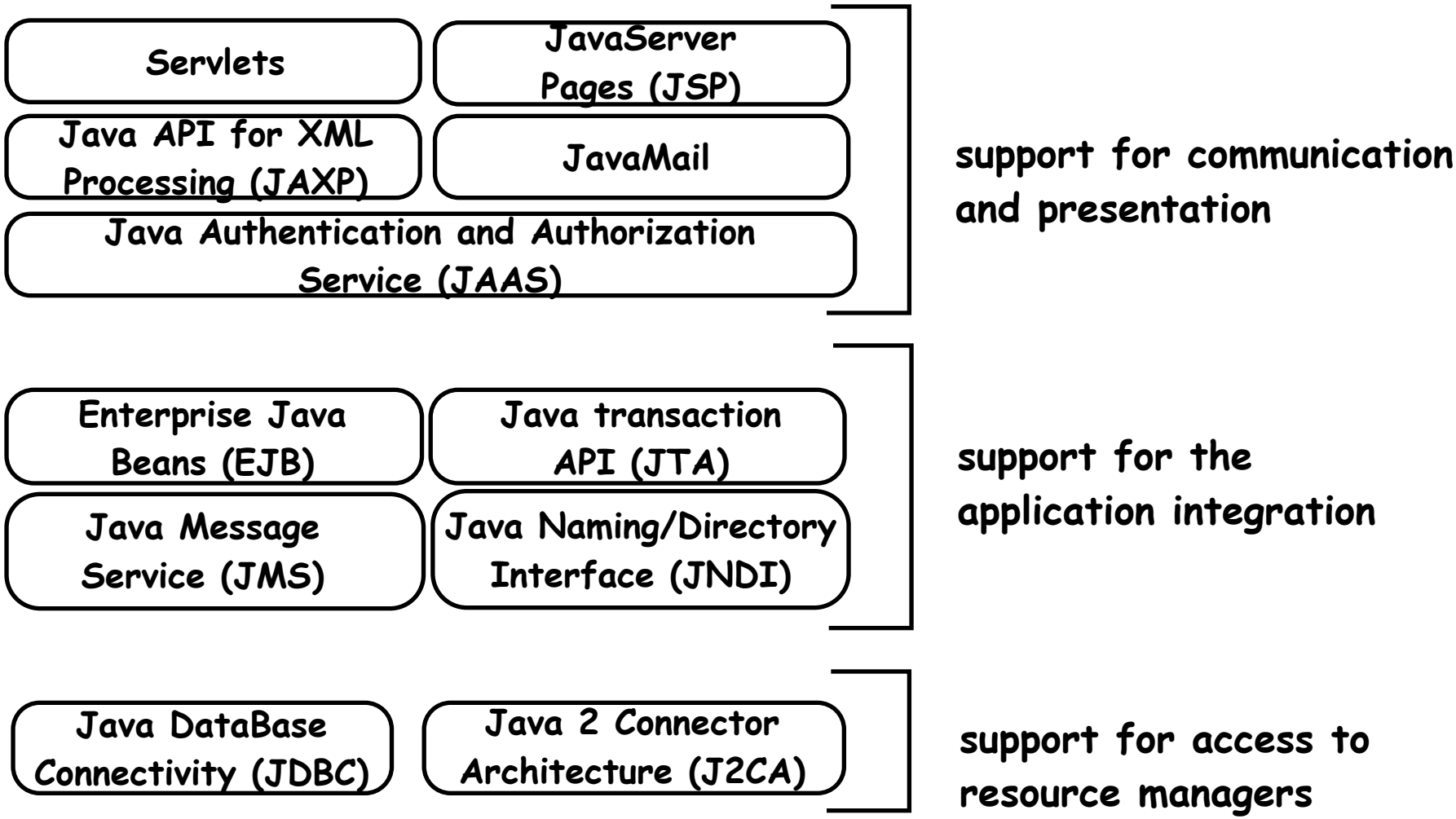
Application Server

- Middleware specialized for Web access
- Presentation layer plays more important role

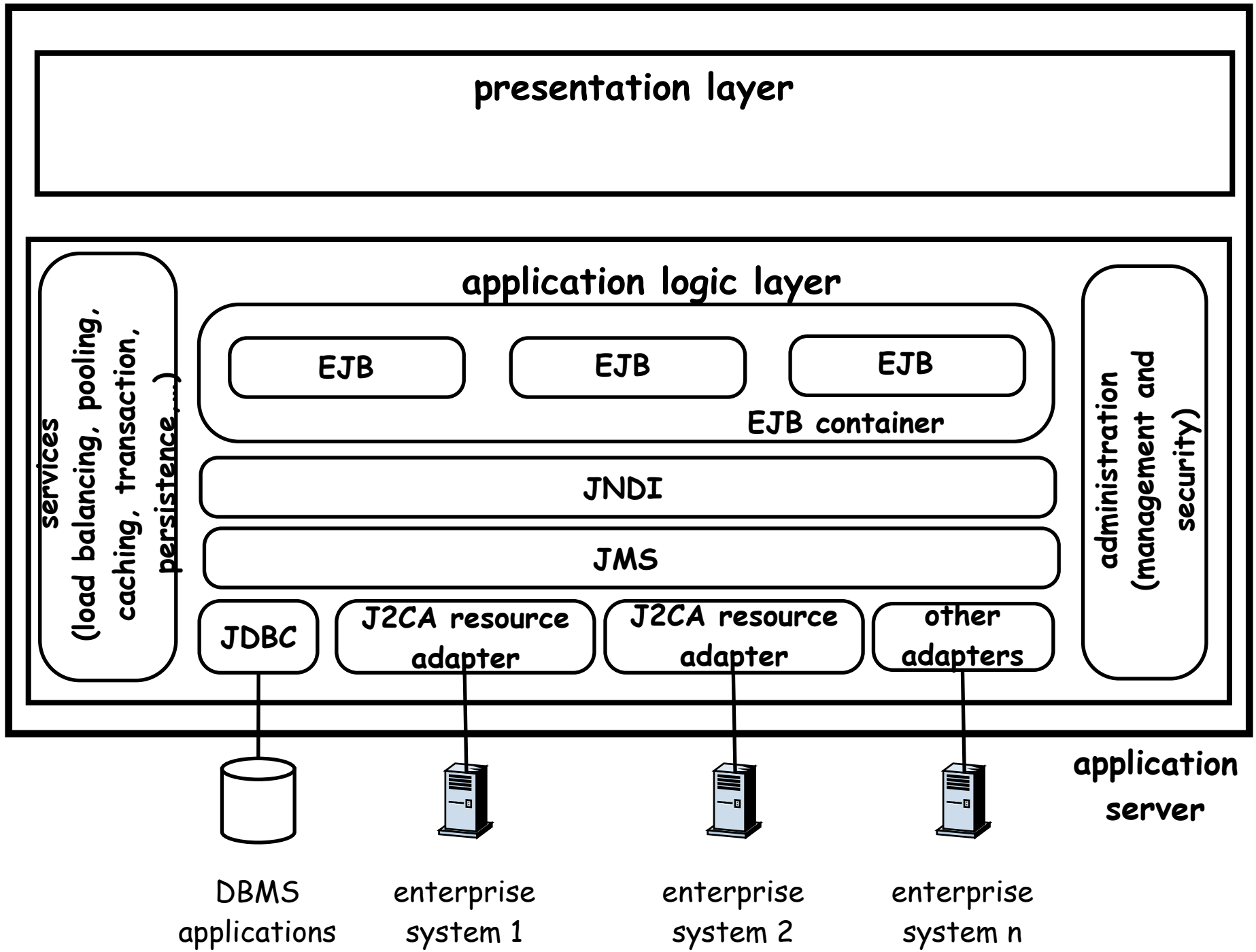
Application Server - General



J2EE Application Server Architecture



J2EE Application Server Architecture - II



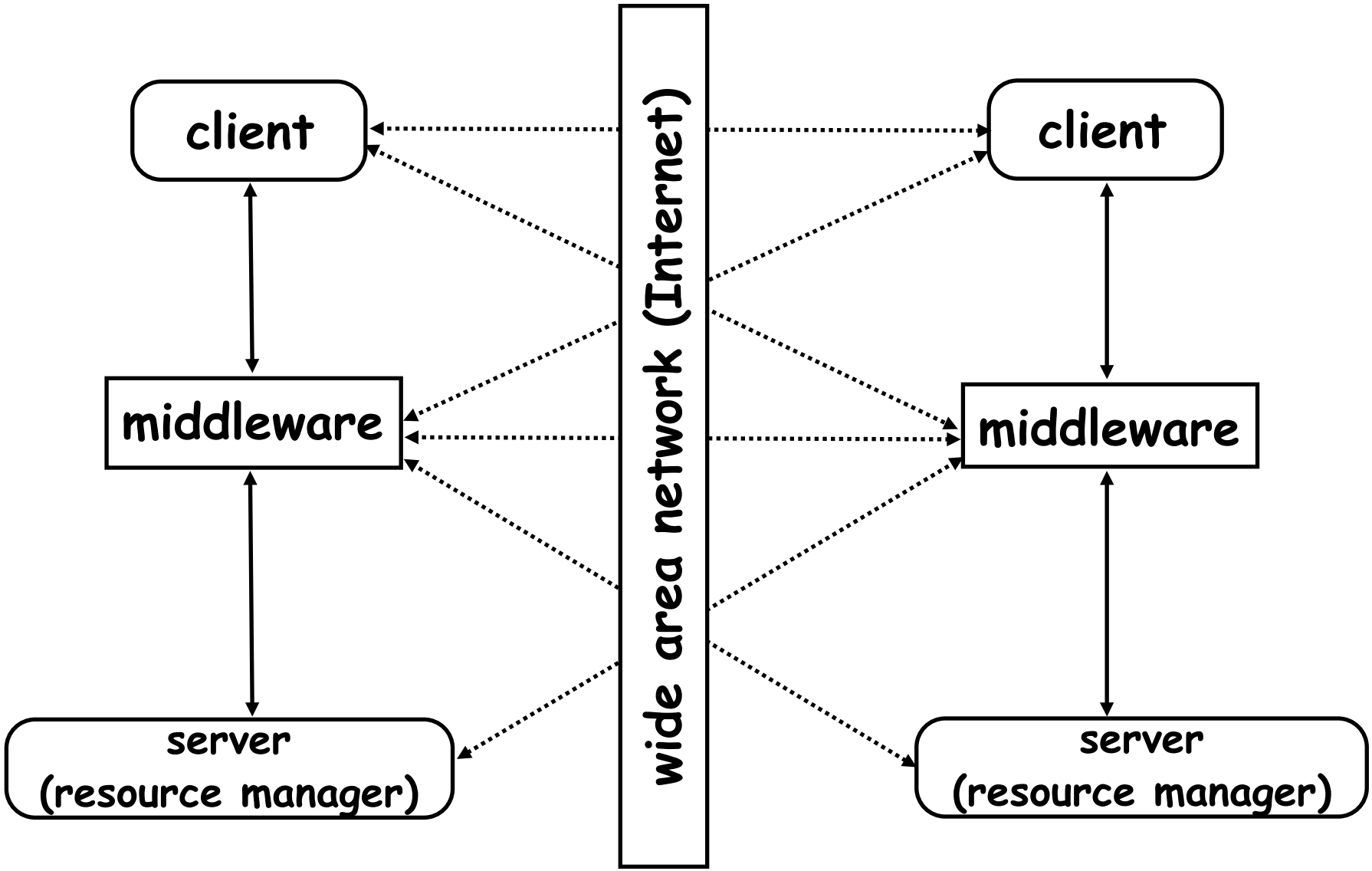
J2EE Application Server Architecture - III


client

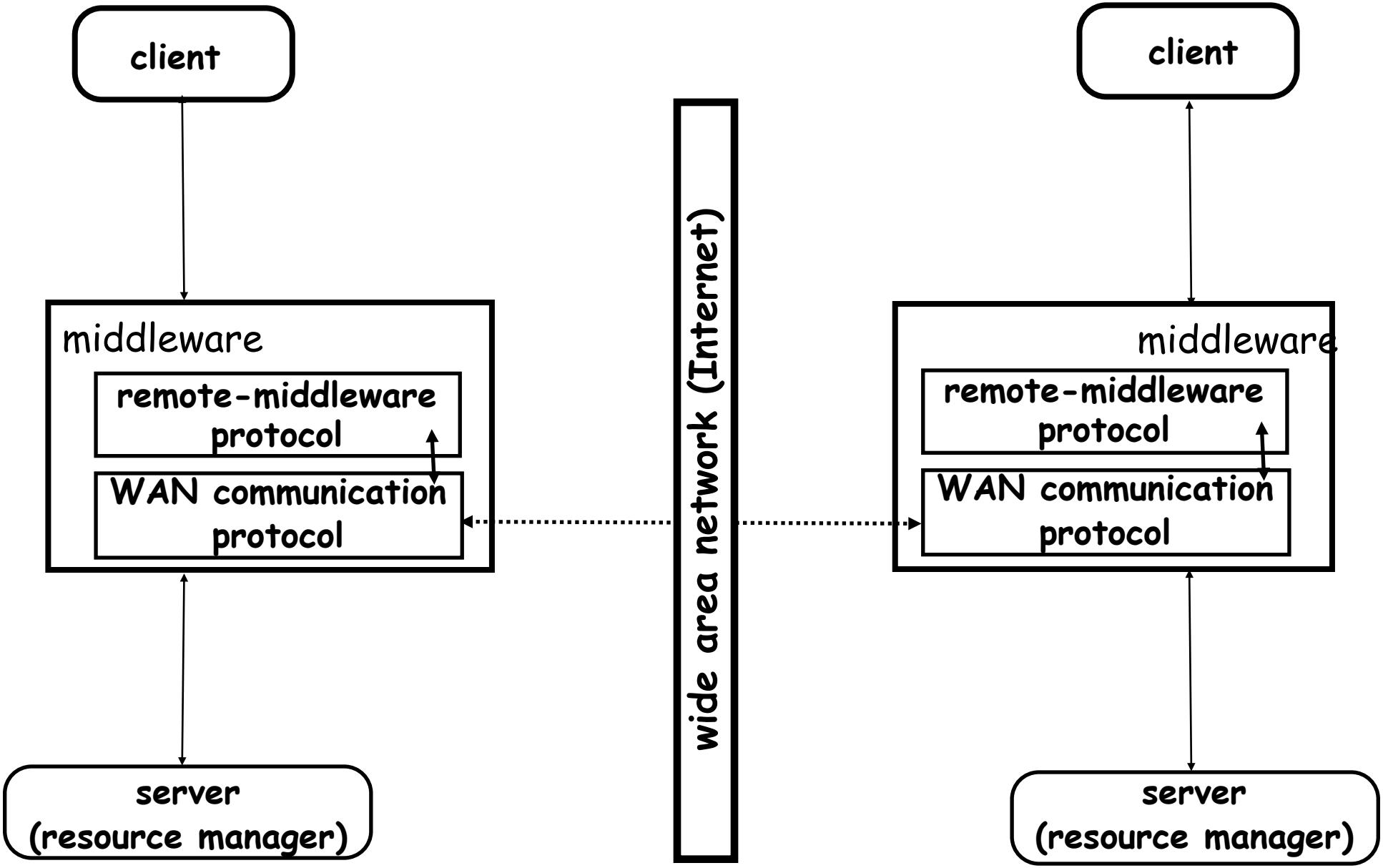


servers for other connections (e.g., WAP)
E-mail server
Web server
services (load balancing, pooling, caching,...)
multidevice content delivery
Servlets
JSPs
XML support
Web services support
personalization logic
administration (management and security)
presentation layer
application logic layer
connection to resource mgmt layer
resource management layer
application server

Wide Area Integration



Wide Area Integration



- Middleware instances need to communicate

Tunneling Through Firewalls