

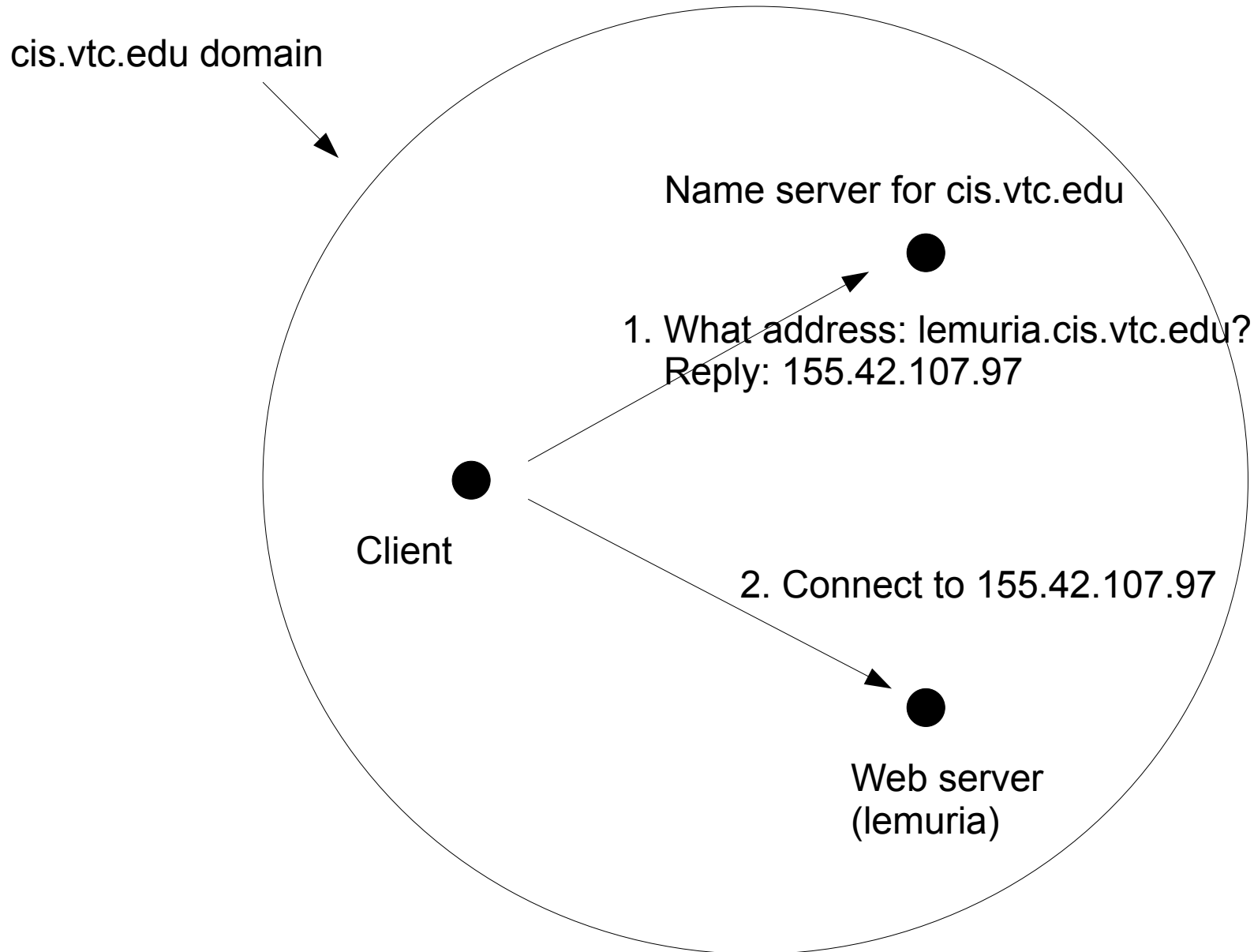
The Domain Name System

Vermont Technical College
Peter C. Chapin

DNS

- The domain name system...
 - Resolves names to IP addresses
 - `lemuria.cis.vtc.edu` to `155.42.107.97`
 - Is a distributed database
 - Each organization maintains its own part.
 - Several fields associated with each domain name.
 - Not just (or not only) IP addresses.
 - Is extensible. New fields can be added.
 - Although this is rarely done.
 - Is described in RFC-1034 and RFC-1035 (plus updates)

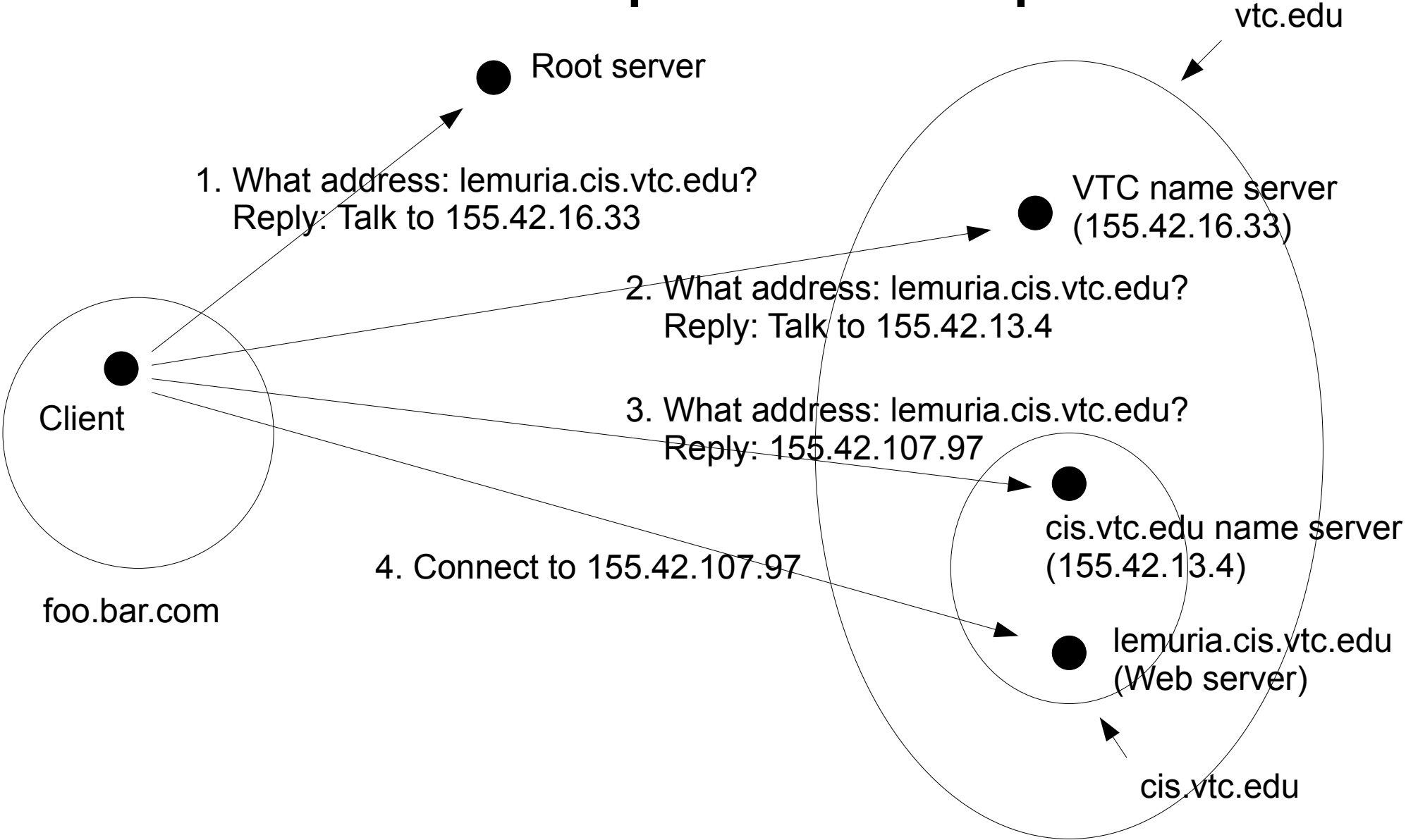
Basic Idea



DNS Resolution

- Client needs to know IP address of name server
 - This is a configuration detail
 - Typically handled with DHCP at client boot time
 - Can be hand specified by user if necessary
- Client sends DNS query
 - Asks for address resolution.
 - Sends name to resolve.
- Name server sends answer
 - Requested IP address or error code.

More Complex Example



Remote Resolution

- Client in another domain...
 - First contacts the “root” name server and asks.
 - Is told the IP address of a name server that might know
 - Root servers know IP addresses of all the “top level” domain servers (e.g. vtc.edu).
 - Contacts next name server and asks again.
 - Is told the answer or referred to a lower level name server.
 - Must have built-in knowledge of root server's IP.
 - Actually there are several root servers spread over the world. IP addresses are published regularly.

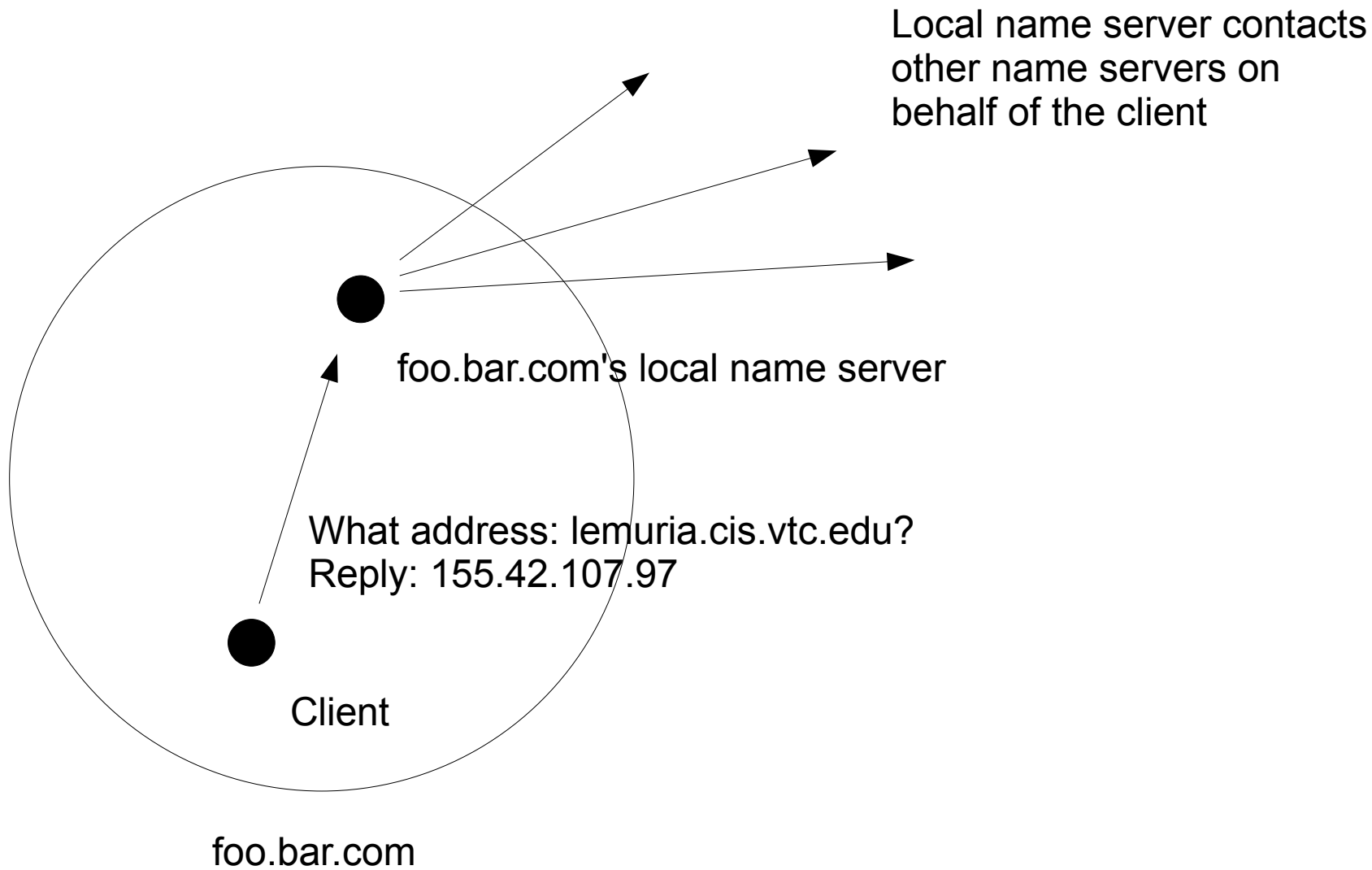
Hierarchy

- This allows a hierarchy of names
 - Name server only needs to know about the names in its domain or... the name servers controlling the subdomains.
 - An organization must register its top level name server(s) with the root servers.
 - “Domain registration”... costs some money.
 - The root servers are incredibly busy
 - Thousands of queries per second!
 - Managed very carefully.
 - Without them, the Internet would fail.

Iterative vs Recursive

- Previous example showed an *iterative* query.
 - Client took responsibility for contacting all necessary name servers.
 - Significant burden to the client.
 - Other clients can't benefit from the results.
- Recursive queries are better.
 - Client asks the local name server to do all the work.
 - Easier for the client.
 - Local name server caches results for other clients.

“Recursive”



UDP and the DNS

- UDP is an excellent protocol for DNS transport
 - Request and reply fit into a single datagram.
 - Request is just a name (+ additional bits)
 - Reply is just an address (+ additional bits)
 - Service is idempotent.
 - Resolving the same name twice produces the same result.
 - We assume that IP address assignments change slowly.
 - Service has no side effects.
 - Resolving a name does not change the state of the server.
 - Low overhead of UDP is very desirable.

DNS Details

- [Cover the specifics of the DNS protocol]