DNS
- DNS elements, services, protocols

DNS: a distributed database providing mapping between Domain name (Hostname) and IP address, an application protocol

Basic functions of DNS
- Low-level name: IP address
- High-level name: hostname

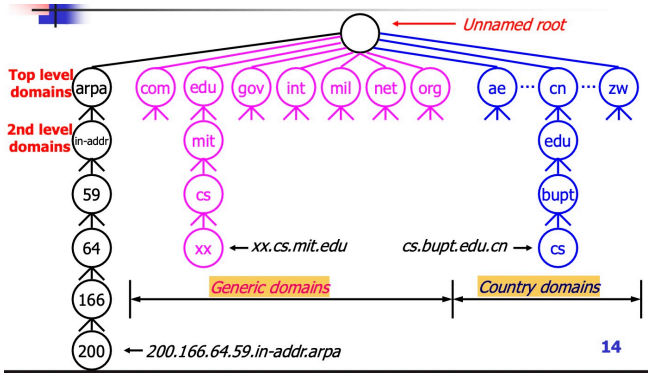
Translating between addresses (Hostname <—> IP address)

Nowadays status of DNS
- Hierarchical structure
- Distributed database
- Efficient, reliable, general purpose

DNS feature
- query/response protocol ruling on top of UDP/TCP, with default port number 53

**Important Terms (DNS elements + DNS services)**

**Hierarchical structure of domain namespace**

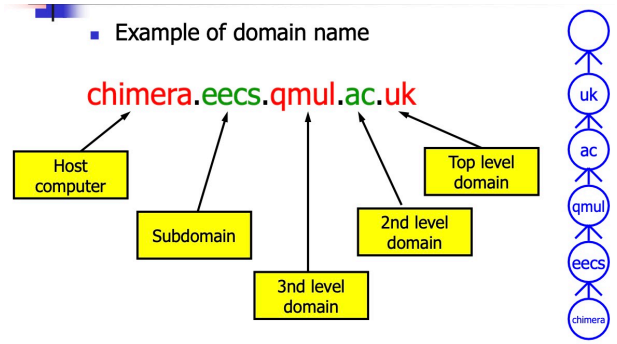



Domain Namespace
Domain namespace: a hierarchical structure like an upended tree

Domain
Domain: each element of the hierarchy
- Root domain: “.”
- Top-level domain (e.g. infrastructure TLD: .arpa —> 反向解析)
- Second-level domain

Domain name
Domain name: the list of the labels on the path from the node to the root of the tree separated by dots (“.”)
- 规则: read left to right (from most specific to least specific — far —> close to root)
- Case insensitive

FQDN: The full name of a domain (Fully qualified Domain Name)



**Resource Record**
Each domain in DNS has one or more Resource Records (RRs)

Resource Records
Resource Records: data associated with a particular name (contain information about that domain)

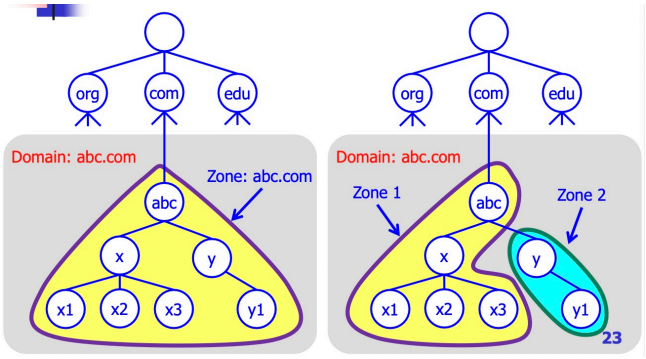
RR contains following information
- Owner — the domain name corresponding to RR
- Type — type of the resource (A, MX, NS, CNAME)
- Class — specifies the protocol family to use (the internet system)
- TTL — specifies the Time To Live of the cached RRs
- RDATA — resource data

Example of resource record

Type=A
- Name= Domain name , Value= IP Address
  *ns.bupt.edu.cn A IN 86400 202.112.10.37*

**Name Servers** (server side of DNS)
Name servers: the repositories (仓库) of information that make up the domain database

Zones
Zone: the data base is divided up into sections, distributed among the name servers.
Relationship zone v.s. domain
- A zone may be one or more domains or even a sub-domain (zone is subset of domain)




Administrative authority
Administrative authority: responsible for that portion (zone) of the hierarchy

Primary server / Authoritative server (official answer)
- Holds in its database the name-to-address mappings for the group of hosts it administers

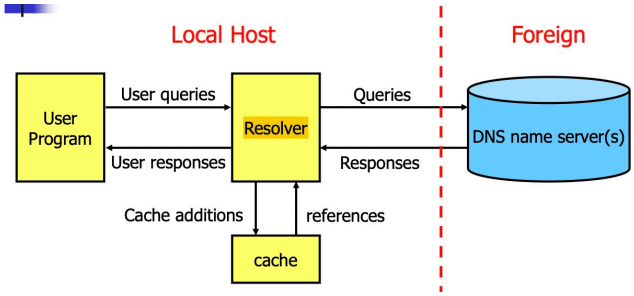
Secondary server
- Maintains a copy of the Primary Server's database

Caching server
- asks DNS queries to other servers but maintains a cache of the responses together with a "time to live" value

**Name Resolvers** (the client side of DNS)
Resolver: the interface between the user program and the domain name servers
Resolver 工作方式:
- Resolver receives a request from a user program
- Asks questions to the DNS system on behalf of the application

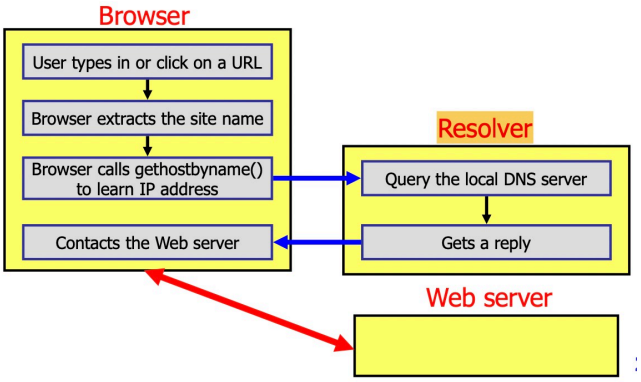
- Returns the desired information

**DNS communication model**




Resolvers are implemented in system calls: gethostbyname(), gethostbyaddr()

**DNS with HTTP connection**

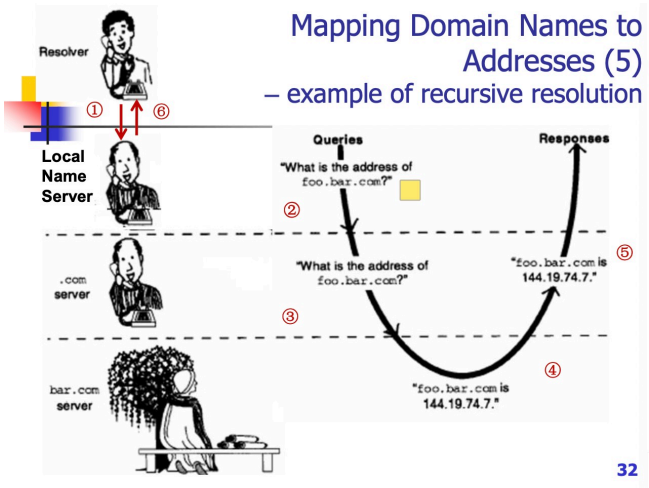


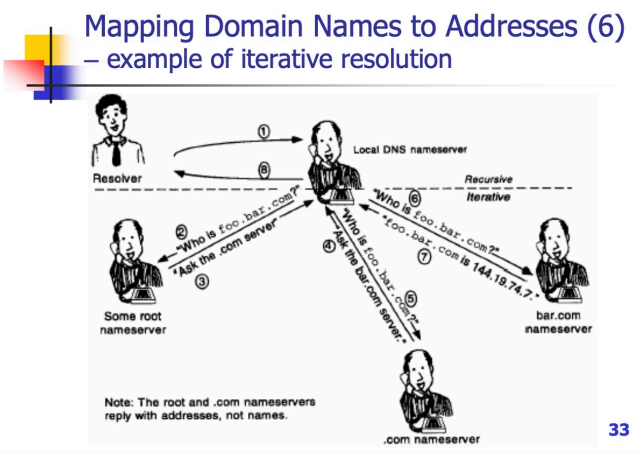
**DNS Services**
- Name resolution
- Query: standard | inverse | pointer

Name resolution (过程/思想)
- Begin with local name server, if cannot resolve a name, sent to another

**Name resolution methods** (2)
Recursive resolution:
If the queried server does not have the information, it must make the appropriate query or queries to get the information (一跳跳去查，再一跳跳返回). A server fulfills a recursive query either with data in its own memory or by making another recursive query. (Local name server给下一个，下一个再给下一个)

Iterative resolution:
If the queried server does not have the information, it may then respond with the address of another server; the local name server then queries that server (都通过local name server去查)




**Inverse Queries & Pointer Queries**
Standard Query
Standard query: mapping a domain name to a resource

Inverse Query (NOT an acceptable method)
Inverse query: mapping a resource to a domain name

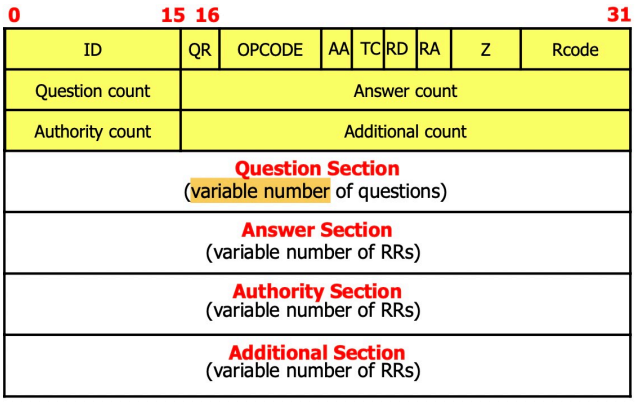
Pointer Queries
Pointer query: using IN-ADDR.ARPA domain for address to host mapping
- Data may be inconsistent

Inverse query v.s. Pointer query
Similarity: IP address (resource) —> domain name
Differences:
- Inverse query: use the same domains as standard query, may need to search the entire set of servers (遍历右边整棵树)
- Pointer query: use IN-ADDR.ARPA domain

**Caching mechanism** to improve efficiency
- Caching at the name server
- Caching at the hosts
- TLD (top level domain) servers typically cached in local name servers —> root name servers not often visited

## DNS protocols

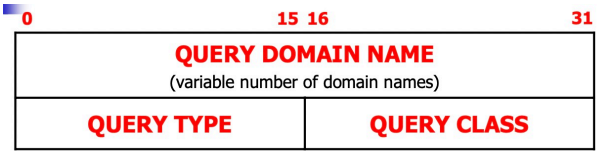
DNS Message Format
Query and Response messages, both with same message format




### Header

QR: query — 0, response — 1
OPCODE: type of query (0 — standard, 1 — inverse, 2)
AA: Authoritative answer
TC: truncation
RD: recursion desired
RA: recursion available
Z: 3-bit
RCODE (0, 1, 2, 3, 4, 5): status of the query — response code
QDCOUNT: number of entries in question section
ANCOUNT: number of resource records in answer section
NSCOUNT: number of name server resource records in authority section
ANCOUNT: number of resource records in additional section

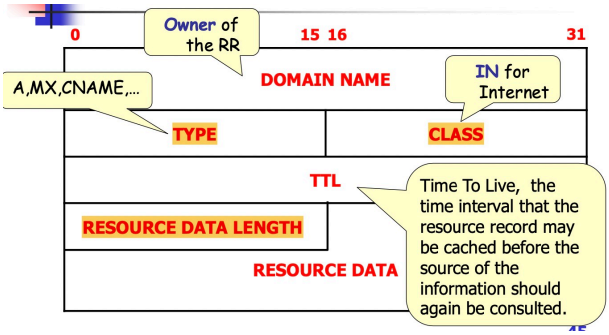
Question Section Format




- QUERY TYPE: 16-bit field used to specify the type of the query
  - A – Host address
  - MX – Mail exchanger for the domain
  - ...
- QUERY CLASS: 16-bit field used to specify the class of the query
  - IN – Internet system
  - ...

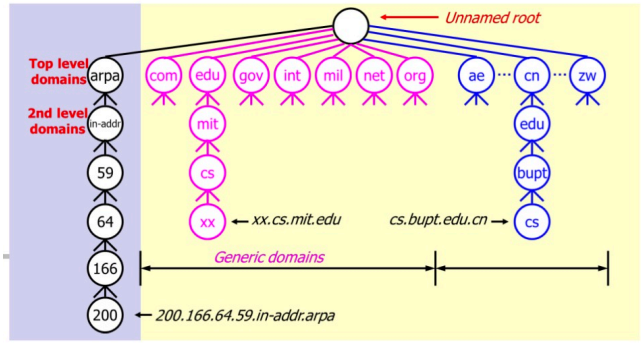
44

Response Section (Answer, Authority, Additional)
Resource Record Format



Type Field
- NS: maps a domain name to the name of a computer that is authoritative for the domain
- A: maps the name to its address. If it has several addresses, separate record for each.
- AAAA: maps the name to its IPv6 address. If it has several addresses, separate record for each.
- CNAME: maps an alias name to the true, canonical name (别名记录)
- MX: Mail exchanger
- PTR: maps an IP address to a system name. Used in address-to-name files

补充：pointer query




200.166.64.59 —> 200.166.64.59.in-addr.arpa (query/search)