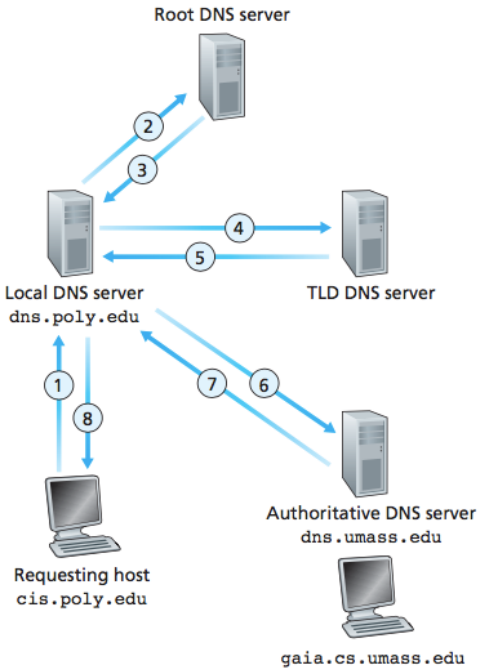
2017-10-25

- **Domain name system (DNS)**:
  - A distributed database implemented in a hierarchy of DNS servers.
  - An application-layer protocol that allows hosts to query the distributed database.
- The DNS protocol runs over *UDP* and uses port 53.
- DNS services: (1) hostnametoIP address translation, (2) host aliasing, (3) mail server aliasing, and (4) load distribution.
- Why not centralize DNS? (1) single point of failure, (2) traffic volume, (3) distant centralized database, and (4) Maintenance.
- The DNS uses a large number of servers, organized in a hierarchical fashion and distributed around the world.
- Three classes of DNS servers: (1) **root DNS servers**, (2) **top-level domain (TLD) DNS servers**, and (3) **authoritative DNS servers**.
- Each ISP has a **local DNS server**, also called **default name server**.
- When a host connects to an ISP, the ISP provides the host with the IP addresses of one or more of its local DNS servers.
- The local DNS server acts a proxy, forwarding the query into the DNS server hierarchy.
- DNS query resolution:
  - Recursive queries: the query asks a local DNS server to obtain the mapping on its behalf.
  - Iterative queries: all of the replies are directly returned to the local DNS server.



- Each time the local DNS server receives a reply from some DNS server, it can cache any of the information contained in the reply.
- The DNS servers that together implement the DNS distributed database store **resource records (RR)**.
- A resource record is a four-tuple that contains the following fields: **(Name, Value, Type, TTL)**.
- The meaning of Name and Value depend on Type:
  - Type A: Name is a hostname and Value is the IP address for the hostname.
  - Type NS: Name is a domain and Value is the hostname of an authoritative DNS server that knows how to obtain the IP addresses for hosts in the domain.
  - Type CNAME: Value is a canonical hostname for the alias hostname Name.
  - Type MX: Value is the canonical name of a mail server that has an alias hostname Name.
- A **registrar** is a commercial entity that verifies the uniqueness of the domain name, enters the domain name into the DNS database, and collects a small fee from you for its services.
- To register the domain name with some registrar, you need to provide the registrar with the names and IP addresses of your primary and secondary authoritative DNS servers.
- The registrar would then make sure that a Type NS and a Type A record are entered into the TLD servers.
- File distribution time:
  - Client/server approach: $\max \{ NF/u_s, F/\min_i d_i \}$
  - P2P approach: $\max \{ F/u_s, F/\min_i d_i, NF/(u_s + \sum_i u_i) \}$
- BitTorrent:
  - **Tracker**: tracks peers participating in torrent.
  - **Torrent**: group of peers exchanging chunks of a file.
- Which chunks to request is determined by **rarest first**.
- Which requests to response to is determined by:
  - **Unchoked**: Top four neighbors that are currently supplying her data at the highest rate.
  - **Optimistically unchoked**: every 30 seconds, one additional neighbor is picked at random and sends it chunks.
- **Distributed hash table (DHT)** is a distributed database that stores the (key, value) pairs over millions of peers. The distributed database will locate the peers that have the corresponding (key, value) pairs and return the key-value pairs to the querying peer.
- DHT has (key, value) pairs, e.g. a key is the content name and the value is the IP address of a peer that has a copy of the content.
- How to assign keys to peers? Given that each peer has an integer identifier and that each key is also an integer in the same range, a natural approach is to assign each (key, value) pair to the peer whose identifier is the *closest* to the key.
- **Circular DHT with shortcuts**:

    - Each peer keeps track of IP addresses of predecessor, successor, and shortcuts.
    - DHT can be designed so that both the number of neighbors per peer as well as the number of messages per query is $O(\log N)$.
    - A satisfactory compromise between the extreme solutions of using mesh and circular overlay topologies.
- To handle peer churn, each peer is required to know the IP address of its two successors.
- NAT prevents an outside peer from initiating a call to insider peer.
- Peers can only communicate through NATs via relay.