- PageRank: absolute score for a page

Architecture of the Web

- biggest info system ever
- success because flexible & incremental
  - layers on top of already functional layers

REST (Representational State Transfer)

- key principles
  - all sources are resources, uniquely accessible via URI
  - client & servers need to know:
    1. URI of resource
    2. Allowed actions (GET, POST, HEAD, ...)
    3. Allowed representations (text/html or others)
  - Client doesn't need to kno how serv gens representation
  - server doesn't need to know how client renders content
  - neither have to be aware of Intermediate proxies
  - there is no HTTP comm state (response doesn't depend on previous req, no side effects)
    - delete/put are unsafe but idempotent
    - get/similar are safe
    - post needs warning, can happen mult times
- typical HTTP reqs:
  - GET data
  - POST (e.g. submit form)
  - PUT (update info)
  - DELETE