

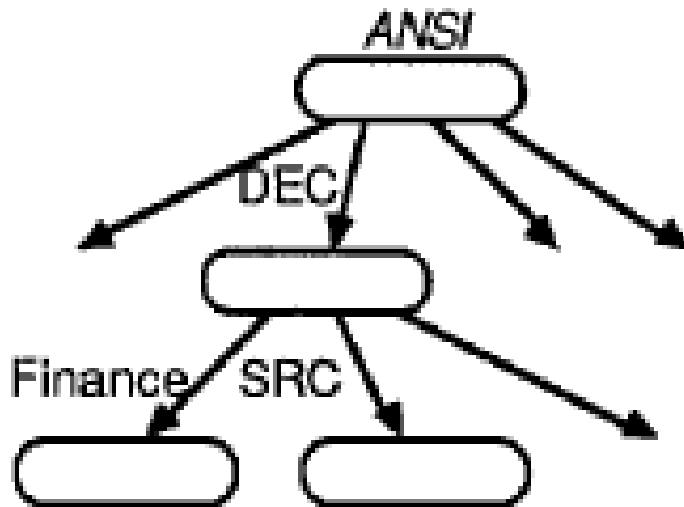


# Designing a Global Name Service

written by Butler W. Lampson,  
presented by Thomas Knauth

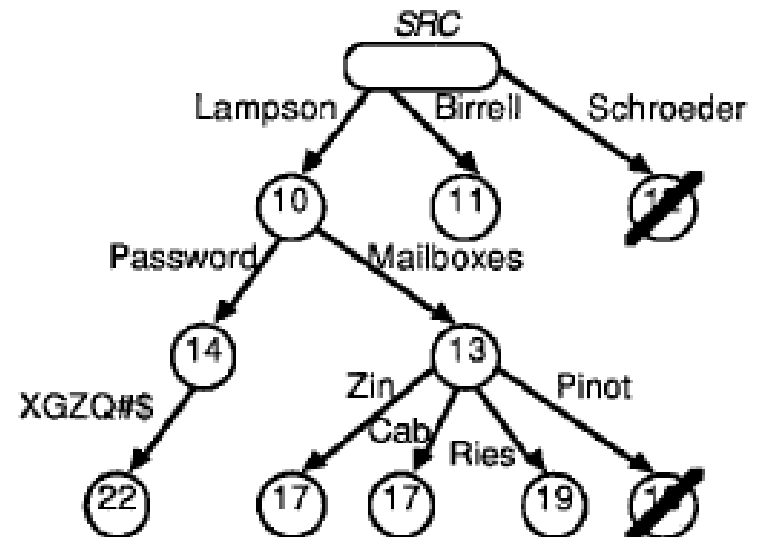
pizza-reading-group, summer term 2007

- What is a name service after all?
  - maps names to values e.g. domain names to IP addresses
- What are the design challenges?
  - size
  - lifetime
  - high availability
  - fault-tolerance
  - mistrust
- Where are name services used today?
  - most prominent is probably DNS
  - in fact, this design uses the name service to look up server names

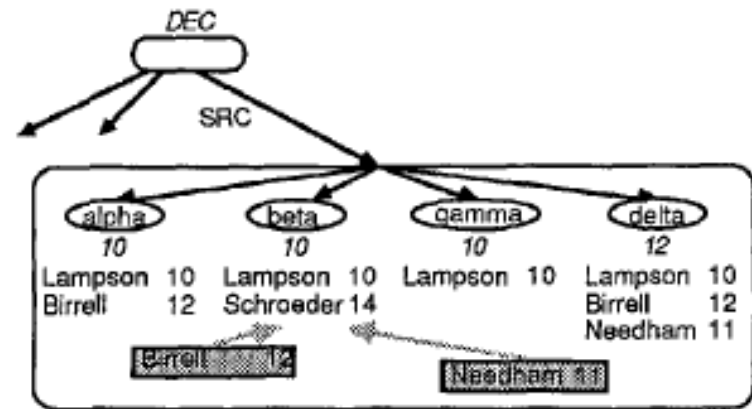


- basic element is the directory
- directory identifier (DI)
- directory reference is value of the name i.e. the DI
- directory maps directory names to directory identifiers

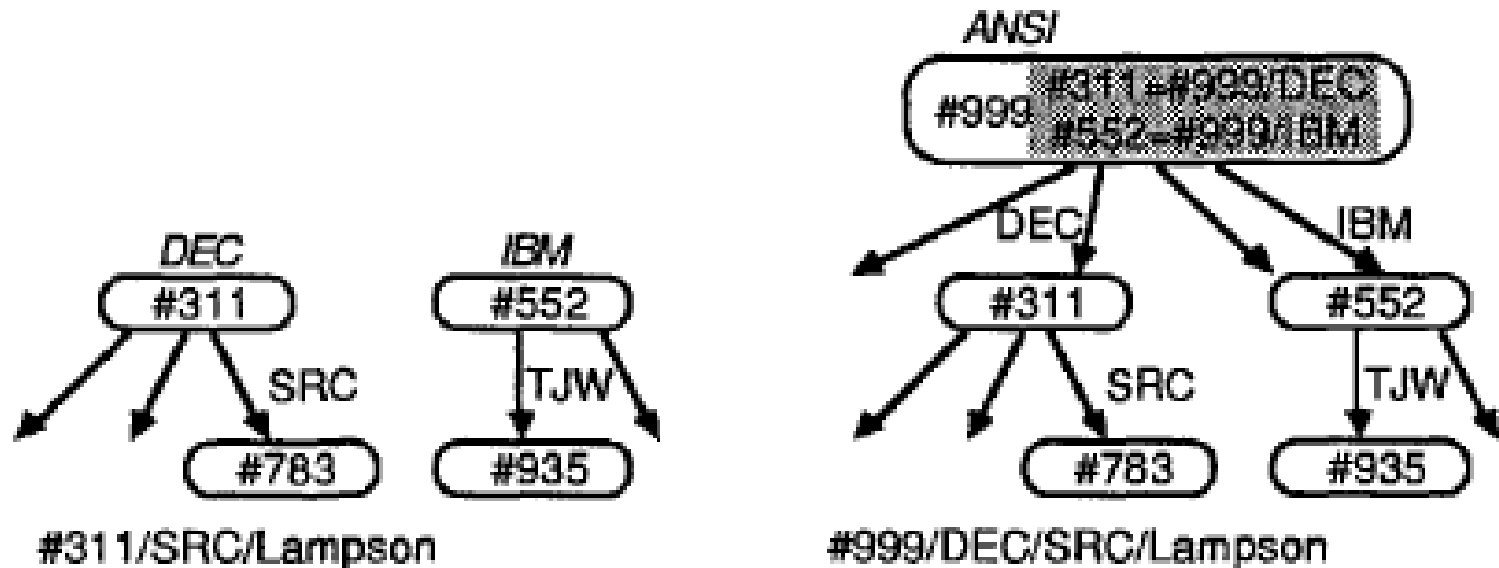
- directory maps names to trees
- labels
- timestamps for synchronization
- mark indicates whether node is present or absent
- value of a path can be
  - single leaf
  - set of leaves
  - sub-tree



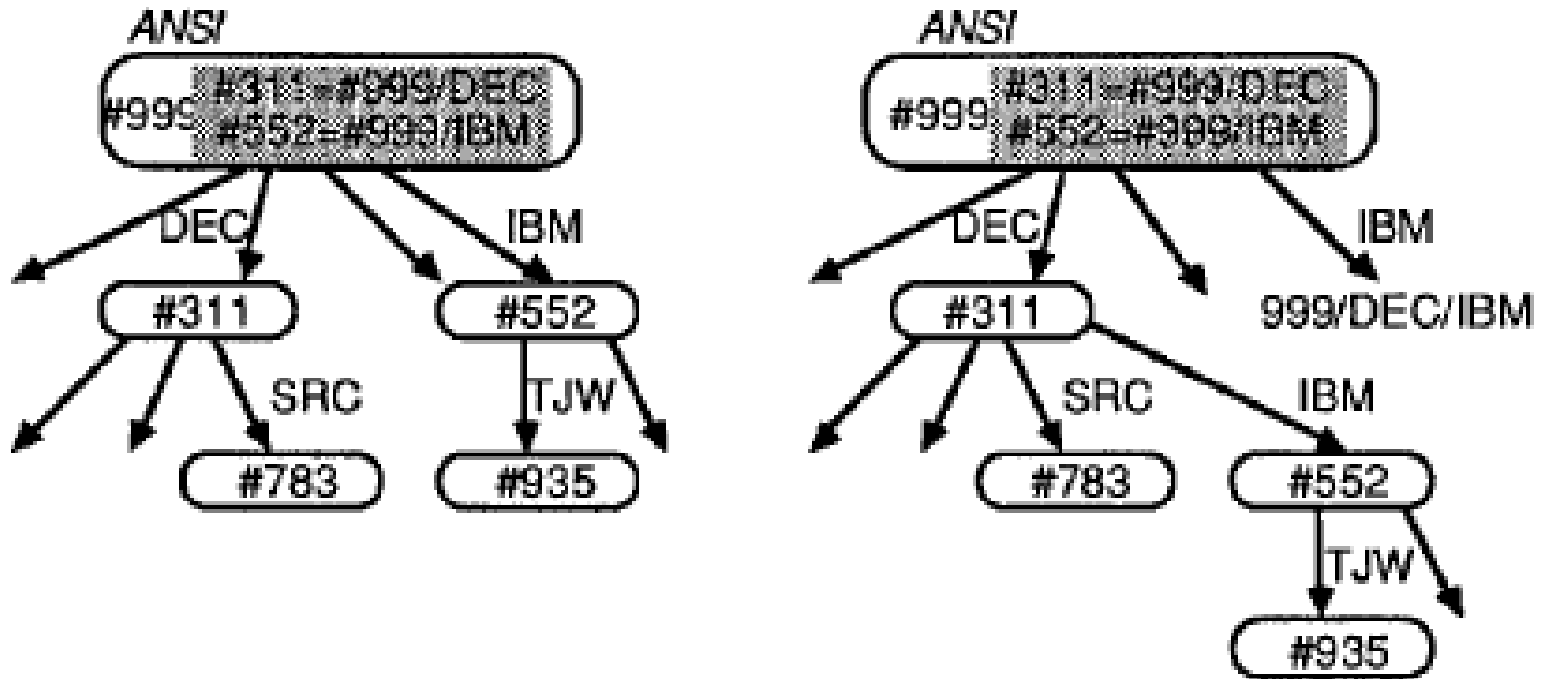
- multiple directory copies
- copies loosely synchronized
  - sync by sweeps, or
  - by messages
- copies linked into a ring



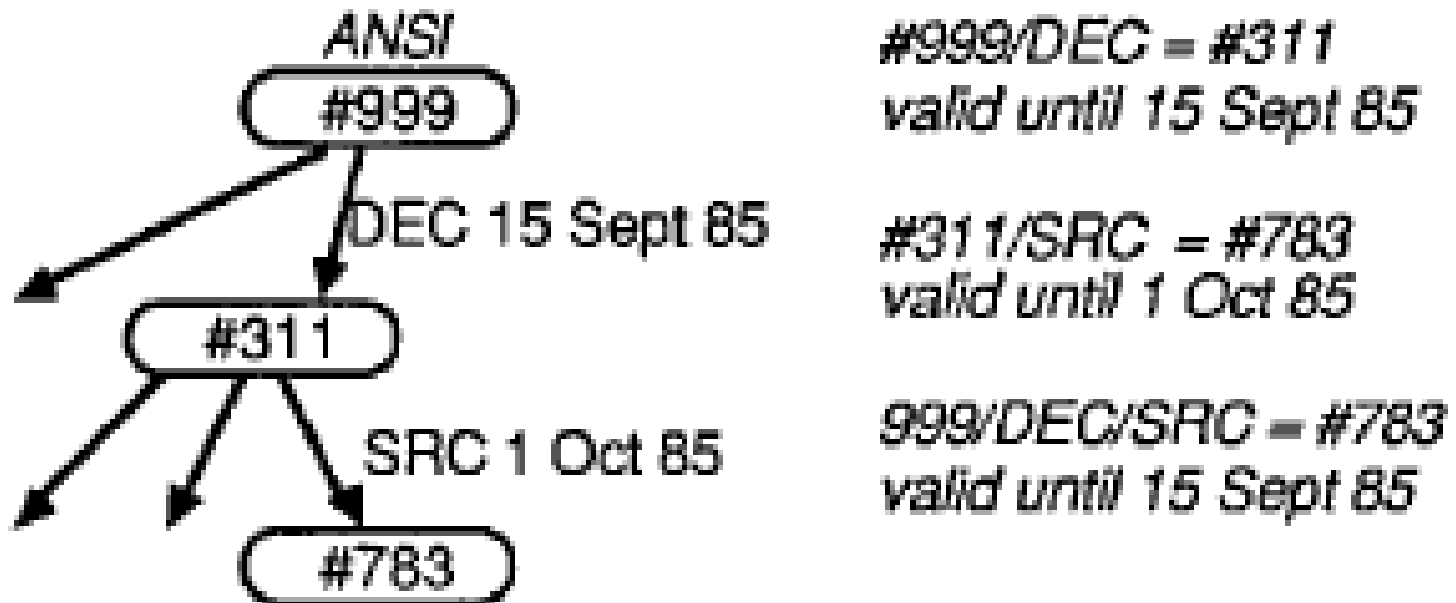
- adding is straightforward
- combining by creating a new root
  - keep set of *well-known* directory identifiers



- making IBM subtree of DEC
  - create a link if old name should still work



- directory reference upper bound on validity called expiration time
- directory lookup can be safely cached for  $t = \text{minimum over all expiration times}$





- intricacies of update/sweeping mechanism
- name service interface
  - client/administrators way to interact with the service
- specification
  - collection of predicates that must hold during execution

- “I expect that most updates will be distributed in messages, but it is extremely difficult to make this method fully reliable. The sweep, [...], is quite easy to implement reliably.” Why?
- What is SRC/Birrell?
  - $I^*/l$  where  $I^*$  is empty? But it's not an only child ...
  - $I^*/l_i$  where  $l_i$  is a leaf node? But not all are leaf nodes ...
  - $I^*$  where the node where it ends is the value of  $I^*$ ; is there a node for  $I^*$  ?!

- Would you expect anything else from a name service? Does technological advance allow for some other design?
- What do current name services (e.g. DNS) look like?
- Has this design proven practical?
- What about multiple roots and links from one tree into another?