Stanford University

Information Technology Systems & Services

Registry & Directory Infrastructure: A Case History

Jeff Hodges

Jeff.Hodges@Stanford.edu
http://www.stanford.edu/~hodges/

14 May 1999

v2.6

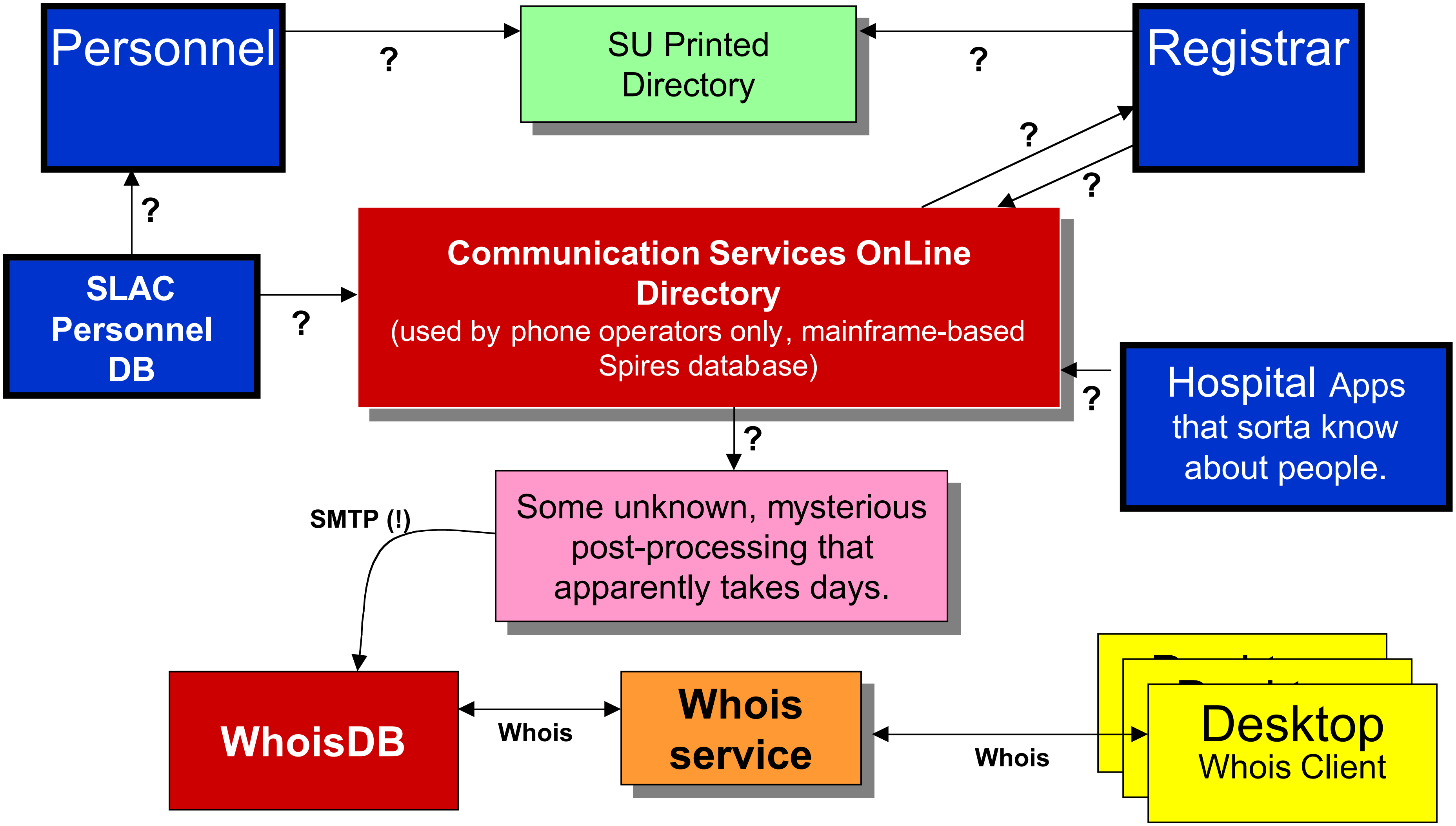
Registry&Directory Infrastructure Overall Scenario

- Stanford is a highly decentralized enterprise with multiple, virtually autonomous, *systems of record.*
- People are a common, shared business object..
  - Personnel’s system: faculty & staff
  - Registrar’s system: students
  - Affiliated enterprises’ systems: SLAC, Hospital
  - Non-trivial number of variously-affiliated people who aren’t in any system
- Other objects: Groups, (network) Services

Registry&Directory Infrastructure Goals

- Support general-purpose whitepages directory service via web-based and teletype user interfaces (UIs)
- Replace (please) our utterly byzantine, amalgamated, and high-latency data feed (next slide).
- Support off-the-shelf products, e.g. browsers with embedded LDAP-based address books, Mail clients, etc.
- Support access by both the Stanford community and the Internet-at-large.
- Push *responsibility* for information maintenance and visibility out to subjects.
- Support & tie-in & leverage-off-of authentication infrastructure -- “SUNet ID”.

Our Data-Feed Challenge



Personnel

SU Printed Directory

Registrar

SLAC Personnel DB

Communication Services OnLine Directory
(used by phone operators only, mainframe-based Spires database)

Hospital Apps that sorta know about people.

Some unknown, mysterious post-processing that apparently takes days.

WhoisDB

Whois service

Desktop
Whois Client

Registry&Directory Infrastructure Overall Problem Statement

- LDAP, and to a lesser degree X.500, are simply *protocols.*
- Though possessing rich informational and functional models, they don't provide any capabilities on their own for, for example, expressing business rules..
  - ensuring any particular identifier is unique
    - [I'm using "identifier" here in terms of an arbitrary attribute value, *not* as a RDN value]
  - ensuring syntax of string-based attribute values
    - This is more an issue with LDAP, X.500 has provisions for server-based attribute syntax validation.
  - etc.

Registry&Directory Infrastructure Definitions and Scope

- Our mission definition for registries..
  - “A *registry* is a service that serves the needs of applications for coordinated maintenance of identity information about a class of business objects.”
    - E.g. Some classes are: People, services, groups.
  - A registry is a transaction-oriented service…
    - Client applications will use one mostly to enter and update information, I.e. a registry is write&update-oriented.
    - Read-oriented access will typically be handled by other components of the overall system, e.g. the Directory.

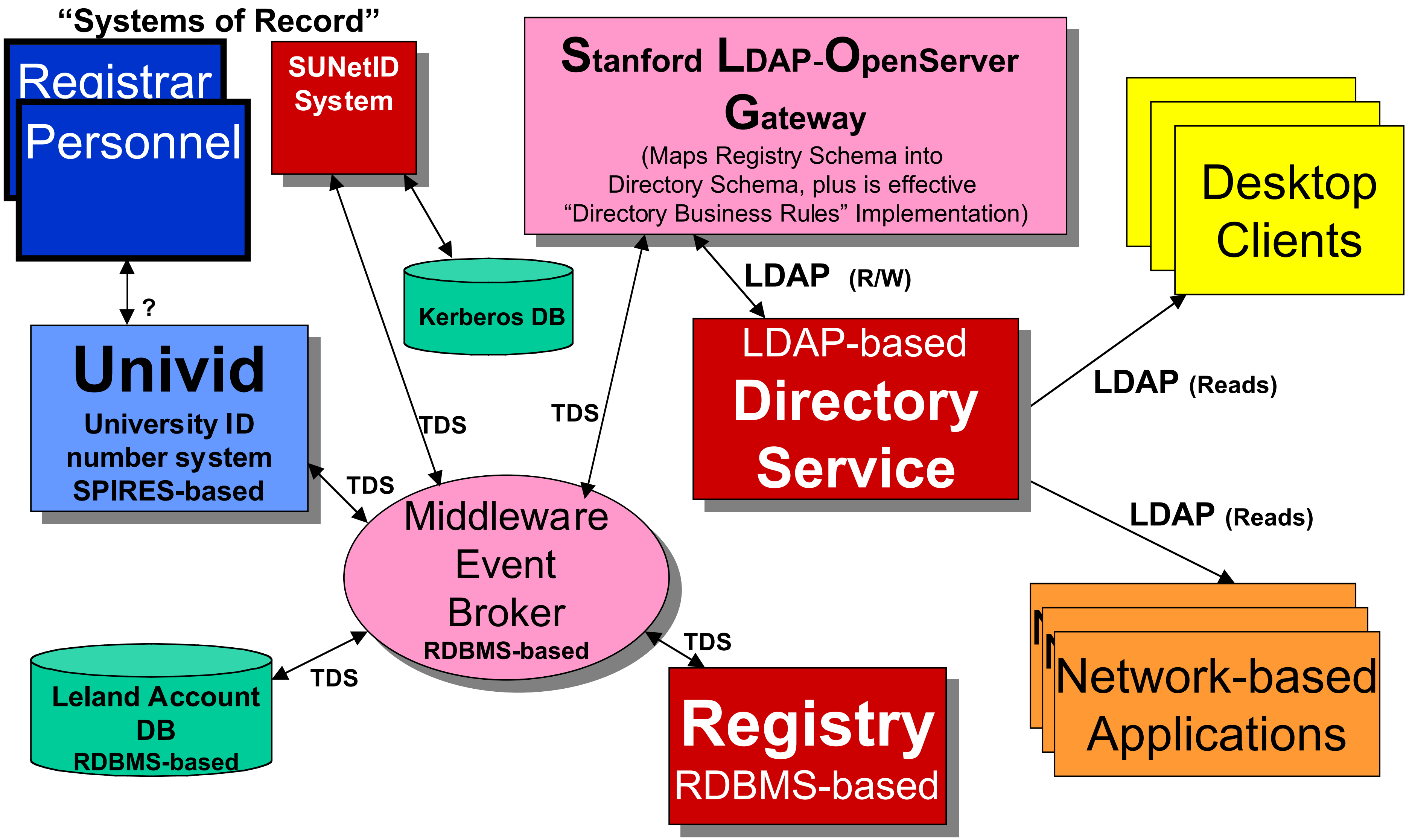
Registry&Directory Infrastructure Definitions and Scope, cont'd

- The scope of our Registry is *enterprise-wide.*
- All people affiliated with the university *should* be in the Registry..
  - I.e. if you need others within the enterprise to recognize your affiliation, you need to be in the Registry.
- A primary materialization of this requirement:
  - Needing an authentication principal - a SUNet ID
  - Many network services are *authenticated*
    - E.g. AFS distributed file system, various web pages, distributed computing resources (e.g. POP-based email service)
    - Authentication infrastructure is Kerberos-based

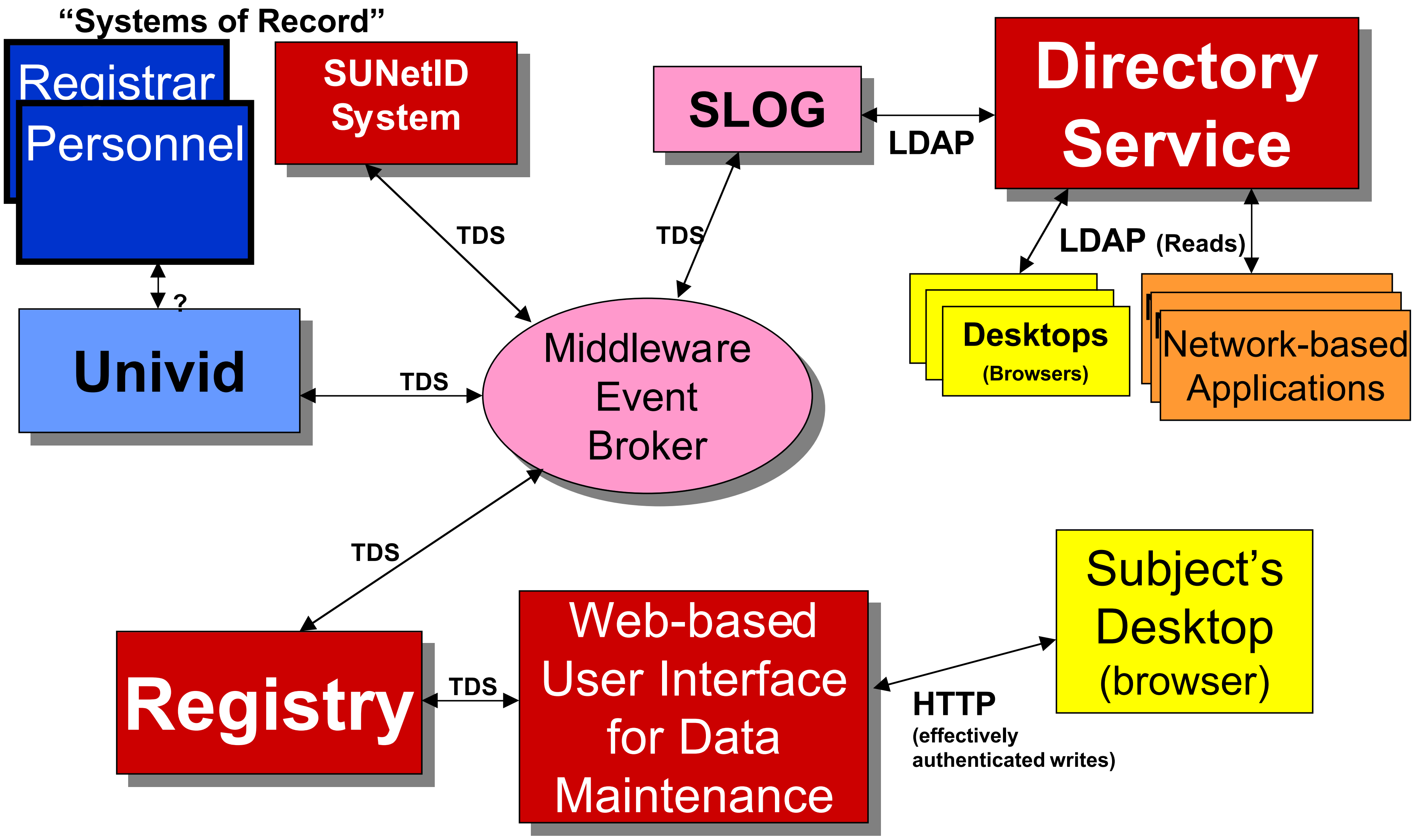
Registry&Directory Infrastructure Definitions and Scope, cont’d

- Registry information is disseminated to other network entities via the Directory
- Various applications utilize the Directory when they need information about people, e.g...
  - “@Stanford.edu” email routing
  - Web authentication
  - Authenticated Printing service
  - Dial-In Network Service
  - Whitepages (I.e. general purpose Directory)
  - HelpSU (“helps-you”) action-request system
  - Rudimentary Authorization Service

Overall Directory&Registry Infrastructure Dissemination



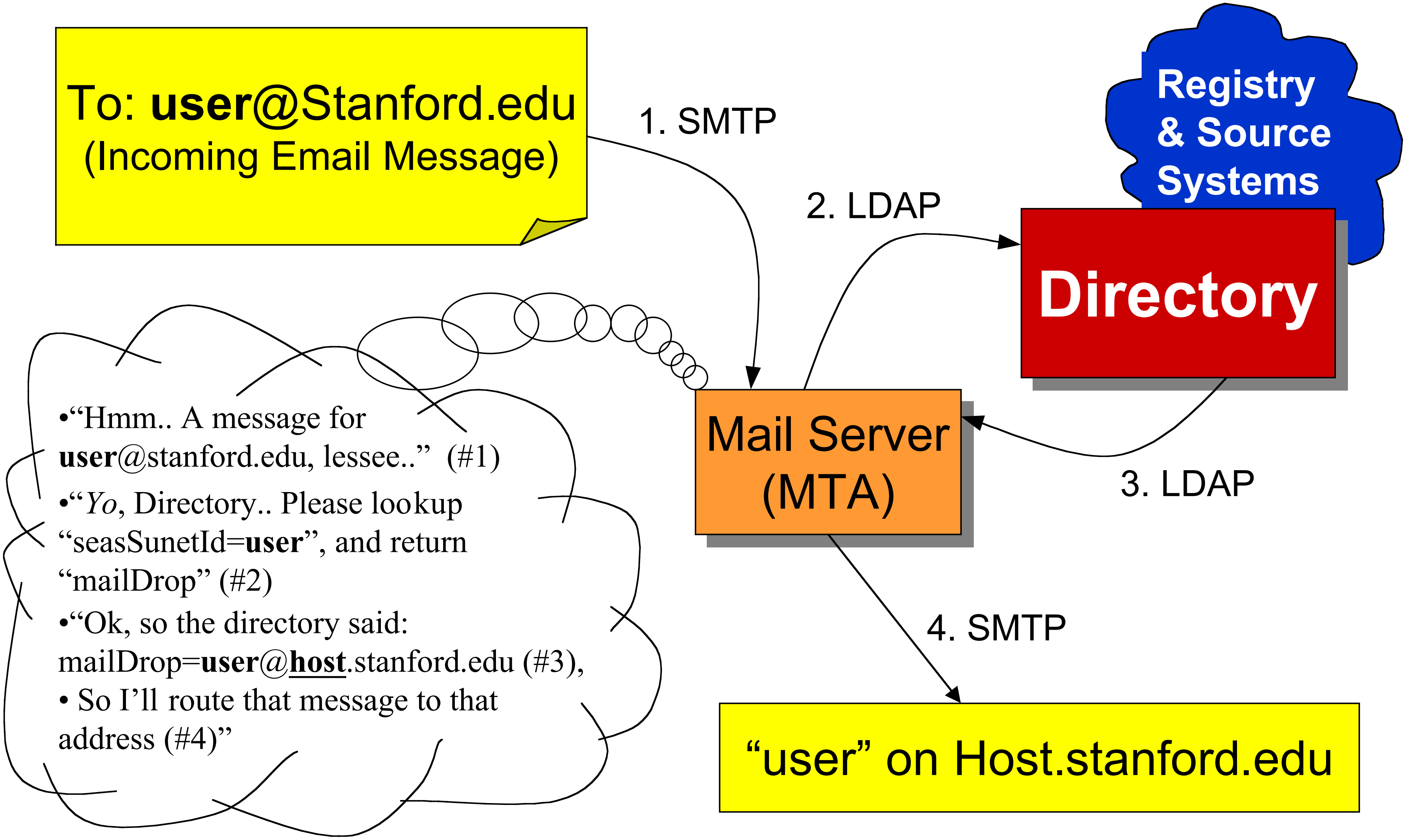
Overall Directory&Registry Infrastructure Update



Registry&Directory Infrastructure
Names and Identifiers

- Names & identifiers are subtlely different beasts
- *Identifiers* are unique
  - A given identifier maps to one subject
  - Subjects have multiple forms of identifiers
  - E.g. “account form” -- 8 chars, alphanumeric
  - “Long form” -- First.Last
  - Some system-specific forms, e.g. Unix UID, DCE UUID, MS GUID, etc.
- *Natural names* are both..
  - Non-unique
  - and *mutable* -- they can & do change
- Use the directory to map all the above to a subject

Registry&Directory Infrastructure
Email Routing



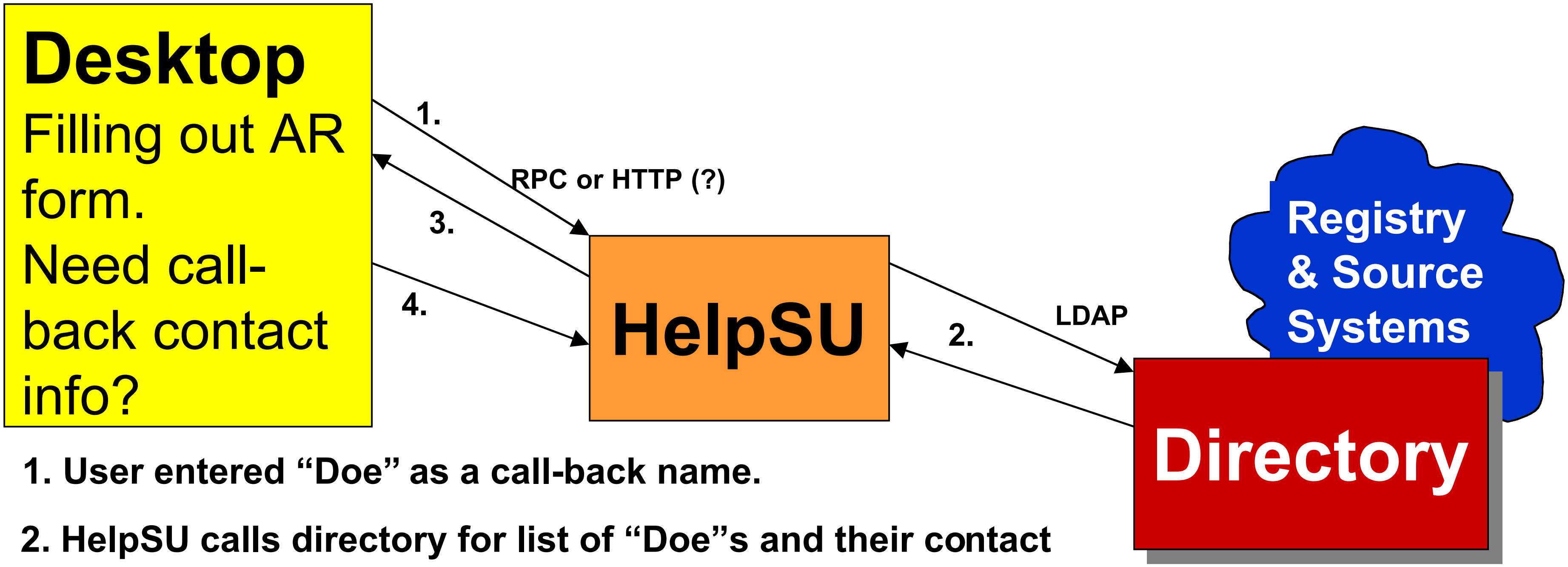
•"Hmm.. A message for **user**@stanford.edu, lessee.." (#1)

•"*Yo*, Directory.. Please lookup "seasSunetId=**user**", and return "mailDrop" (#2)

•"Ok, so the directory said: mailDrop=**user**@**<u>host</u>**.stanford.edu (#3),

• So I'll route that message to that address (#4)"

Registry&Directory Infrastructure HelpSU Action Request System



1. **User entered “Doe” as a call-back name.**

2. **HelpSU calls directory for list of “Doe”s and their contact info.**

3. **HelpSU builds a “pick list” in it’s UI using the directory info for “Doe”.**

4. **User picks person they really meant from the list, and it is entered into the action request.**

Registry&Directory Infrastructure Summary

- The natures of Registries and Directories are subtly different
- X.500/LDAP-based directory services are not RDBMSs
- Makes sense to combine them into an overall system -- play on their strengths
- Project at Stanford is far from, if ever, “finished” -- will continually evolve
  - Present deployment effort is “Phase II”
  - Phase III will involve policy implementation in order to support off-the-shelf LDAP clients

Registry&Directory Infrastructure Themes / Philosophies

- DNs are immutable and persistent
  - a DN is a *primary key,* yet another *identifier.*
  - DNs are *not* necessarily human-palatable.
  - "(Natural) Names" are another class of a subject's attributes and ***aren't*** primary keys in and of themselves.
- A Directory can't do it *all* by itself
  - Has to have site-specific procedures & conventions wrapped around it.
  - E.g. how are subjects vetted and assigned their initial identifier?
  - E.g. selecting a unique identifier of some given form.

Registry&Directory Infrastructure Themes / Philosophies, cont'd

- How effectively user-oriented applications can leverage off of a directory infrastructure is *directly proportional* to how well-formed and well-specified the system's notions of *identifiers* & *names* are.
- The currently prevalent directory access protocols, *in and of themselves*, are *not* "strong" authentication protocols.
- Directory technology is a key underlying enabler for Authorization Services (among lots of other possibilities).
- Like the "single-sign-on" notion morphing into "fewer-sign-ons", the "single directory repository per administrative domain" notion should more realistically be "fewer repository/directories, with cleanly-crafted roles and data feeds".

Registry&Directory Infrastructure Themes / Philosophies, cont'd

- "Trust management" is an often used term, but perhaps a more relevant way to consider the notion is as its inverse..
  - "*Risk Management*".
- Privacy is a huge, emerging issue.
  - We all need to pay a lot more attention to it.
- "Open is good, closed is bad"
  - where protocols, and to a large extent even implementations, are concerned.
- All the above is IMHO, of course.

Acknowledgements

- The Registry & Directory (recently subsumed into “The Horton Project”) team is comprised of (at least):
  - Booker Bense, Carol Farnsworth, Jill Fukuhara, Michael Hart, Jeff Hodges, Craig Jurney, John Klemm, Bill Lucker, Jeff Mapes, Danno McKinnon, Lynn McRae, Dennis Michael, RL “Bob” Morgan, Catherine Mulhall, Pat Nolan, Michael Puff, Dennis Rayer, Sandy Senti, Tim Torgenrud, Dwayne Virnau.
- Lynn McRae (leader/manager), Craig Jurney, Michael Puff, Michael Hart, with RL “Bob” Morgan (enterprise Architect) largely conceived of and implemented the Registry, with input from the team at-large.

Acknowledgements, cont'd

- Jeff Hodges (Tech leader), Danno McKinnon, and Tim Torgenrud (manager) configured and deployed the directory, designed the schema (with input from the Registry team folk), and designed and implemented the Access Control approach, again with overall input from the team at-large.
- RL "Bob" Morgan authored the initial drafts of the SUNetID concept and was generally responsible for evangelizing our (still emerging & evolving) concept of a single identifier namespace.

References

- This talk will be available at..
  - http://www.stanford.edu/people/hodges/talks/
- Selected References..
  - Stanford Registries & Directories pages..
    - http://www.stanford.edu/group/itss-ccs/project/registry/
    - http://www.stanford.edu/group/itss-ccs/project/registry/registries.html
    - http://www.stanford.edu/group/itss-ccs/project/sunetid/
    - http://www.stanford.edu/group/networking/directory/
    - http://www.stanford.edu/group/networking/directory/models/Word_Dir_Svcs_Model_10-29-98-edited-jdh/Word_Dir_Svcs_Model_10-29-98-edited-jdh.htm
  - Project Horton
    - http://www.stanford.edu/group/itss-ccs/project/horton/
  - SUNet ID
    - http://www.stanford.edu/group/itss-ccs/project/sunetid/

References, cont'd

- *Why do I need a Directory when I could use a Relational Database?* -- by Steve Kille
  - http://www.stanford.edu/~hodges/talks/EMA98-DirectoryServicesRollout/Steve_Kille/index.htm
- *Directory Services: DIT Design -- James Rommel*
  - http://www.stanford.edu/~hodges/talks/EMA98-DirectoryServicesRollout/RommelEMA/index.htm
- *Risk Management is Where the Money Is*
  Dan Geer, CertCo
  - http://www.stanford.edu/~hodges/doc/Geer-RiskManagement.txt