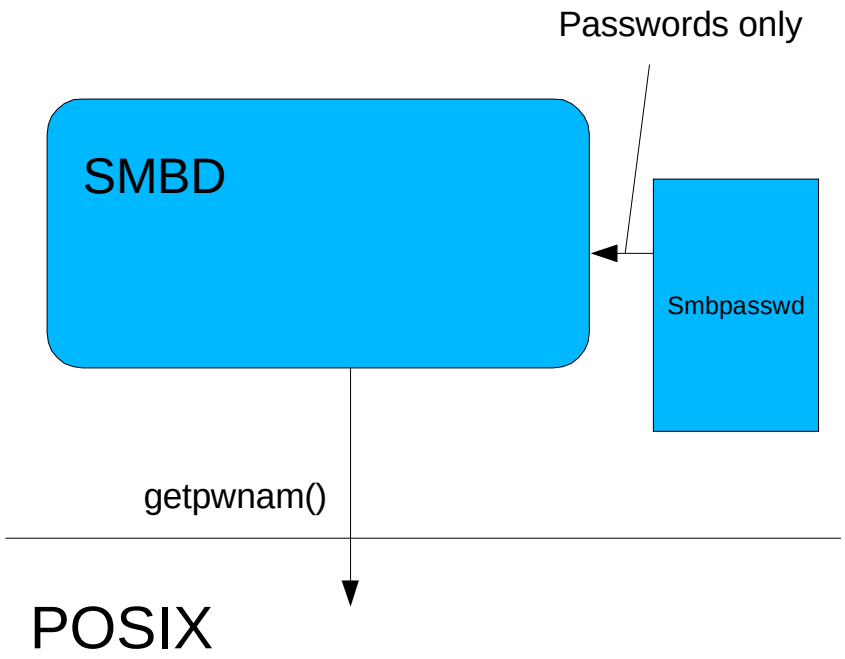
SAM, GUMS, IDMAP
From discussion to reality

by
Andrew Bartlett & Simo Sorce

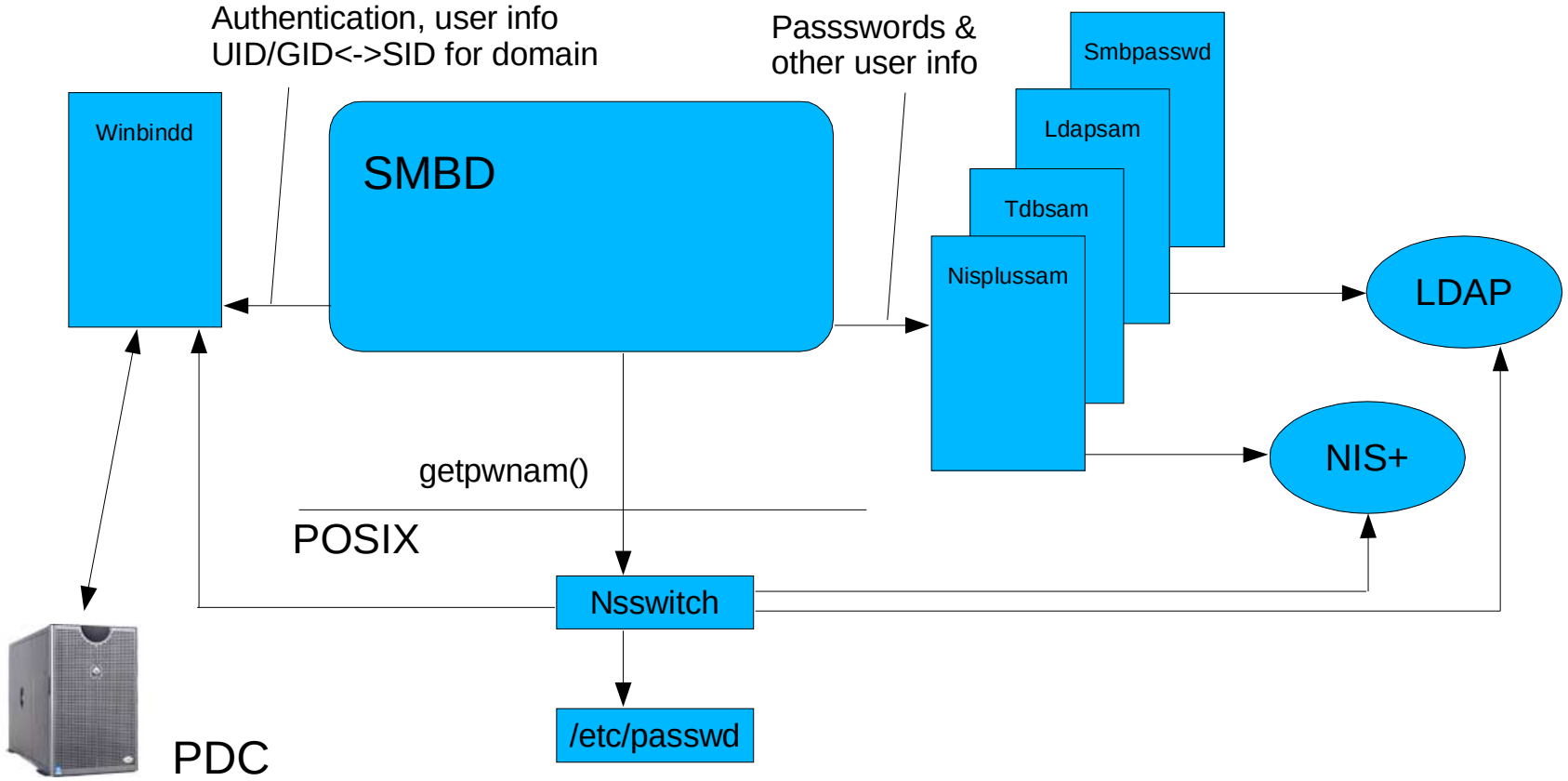
User Management in samba 2.0

- Smbpasswd is used mainly for password storage
- No other password database backend
- Smbpasswd stores the unix user name and uid

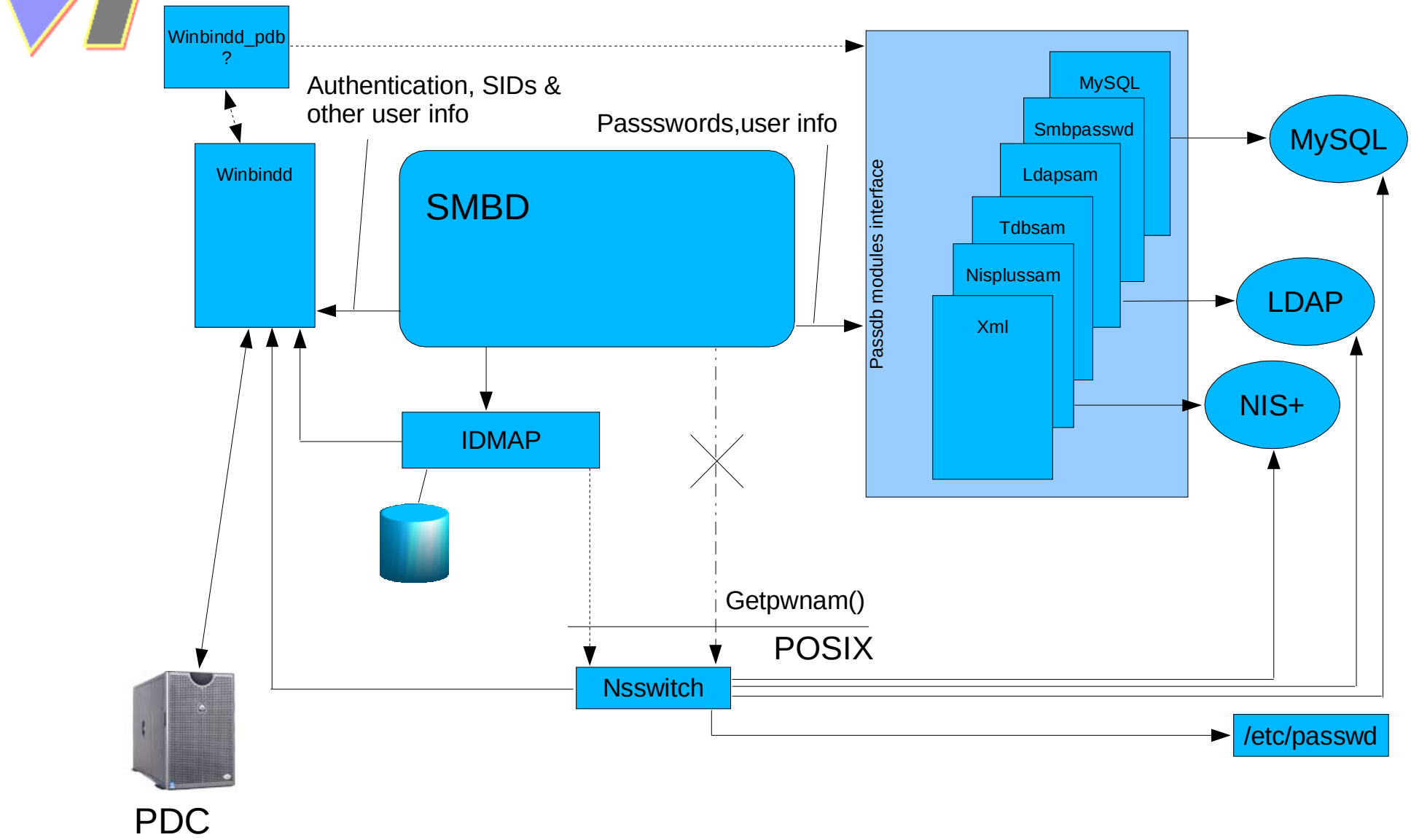


User Management in samba 2.2

- Multiple password databases
- Databases can store other Windows related user data
- The backends store the unix user name and the uid
- Domain users provided through winbindd



User Management in samba 3.0alpha



What is a 'SAM'

- Users:
  - Username
  - Full Name, Description
  - SID
  - Password
  - Home, Profile, ... locations
  - Logon restrictions
    - Hours
    - Machines
    - Expiry
  - 'Times'
  - Dialup Properties
- Machines, Trusted Domains...

Our passdb

- Loadable modules
- Weak group support
- No privileges support
- Arbitrary RID support

- Passdb
  - Smbpasswd
    - Stores only passwords
  - Tdbsam
    - Stores all the user informations as NT4 does
    - Easy to set up
    - Easy to back-up through tdbdump

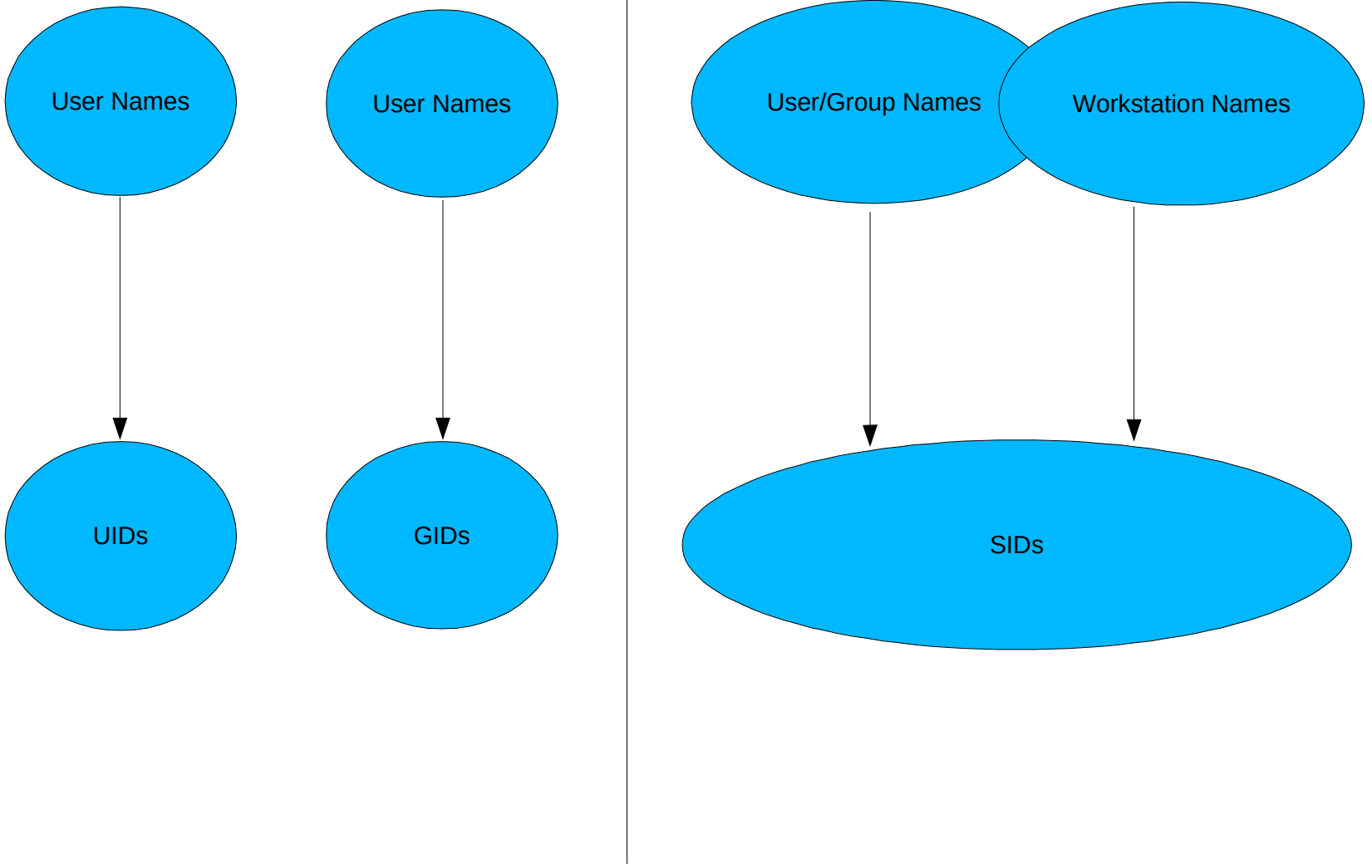
Our passdb

- Ldapsam
  - Stores all the user informations as NT4 does
  - Easy Unix/Samba user information coupling
  - Easy replication over multiple servers
  - Easy multi-DC/multi-Server infrastructures
  - Not so easy to setup for non-experienced admins
  - Easy integration with other services (Mail, ...)

So where is the problem?

- Windows uses Security IDs (SID), not UIDs or GIDs.
- A SID can identify more things than merely users or groups
  - World (S-1-1-0)
  - Local System (S-1-5-18)
  - A domain (S-1-5-21-1721414241-570541885-638950510)
  - All authenticated users (S-1-5-11)
  - ...
- Windows have a unified case-insensitive name space.
- NT Local Groups can contain groups and users
- Posix groups can contain only users.

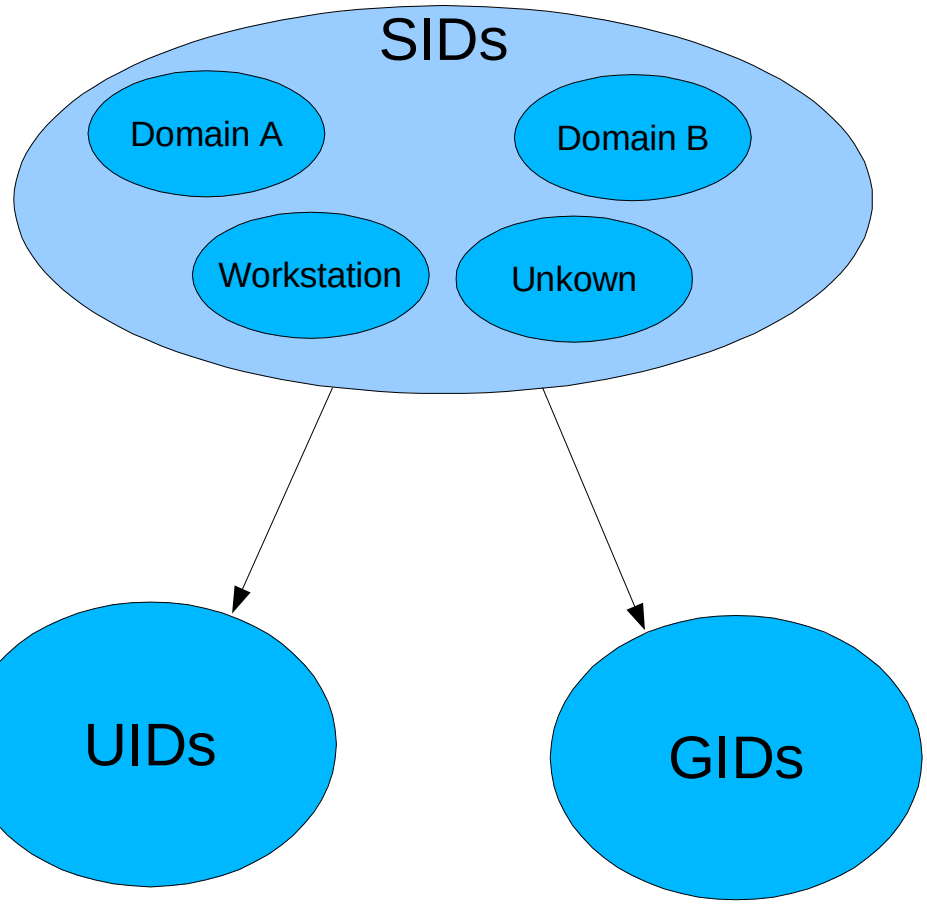
Names and ID spaces



The Ideal SAM

- Only SIDs no UID/GIDs
- Unified case-insensitive name space
- Never check unix users
- Trust the idmap system
- Possibly users are provided back to the underlying system through winbindd

IDMAP

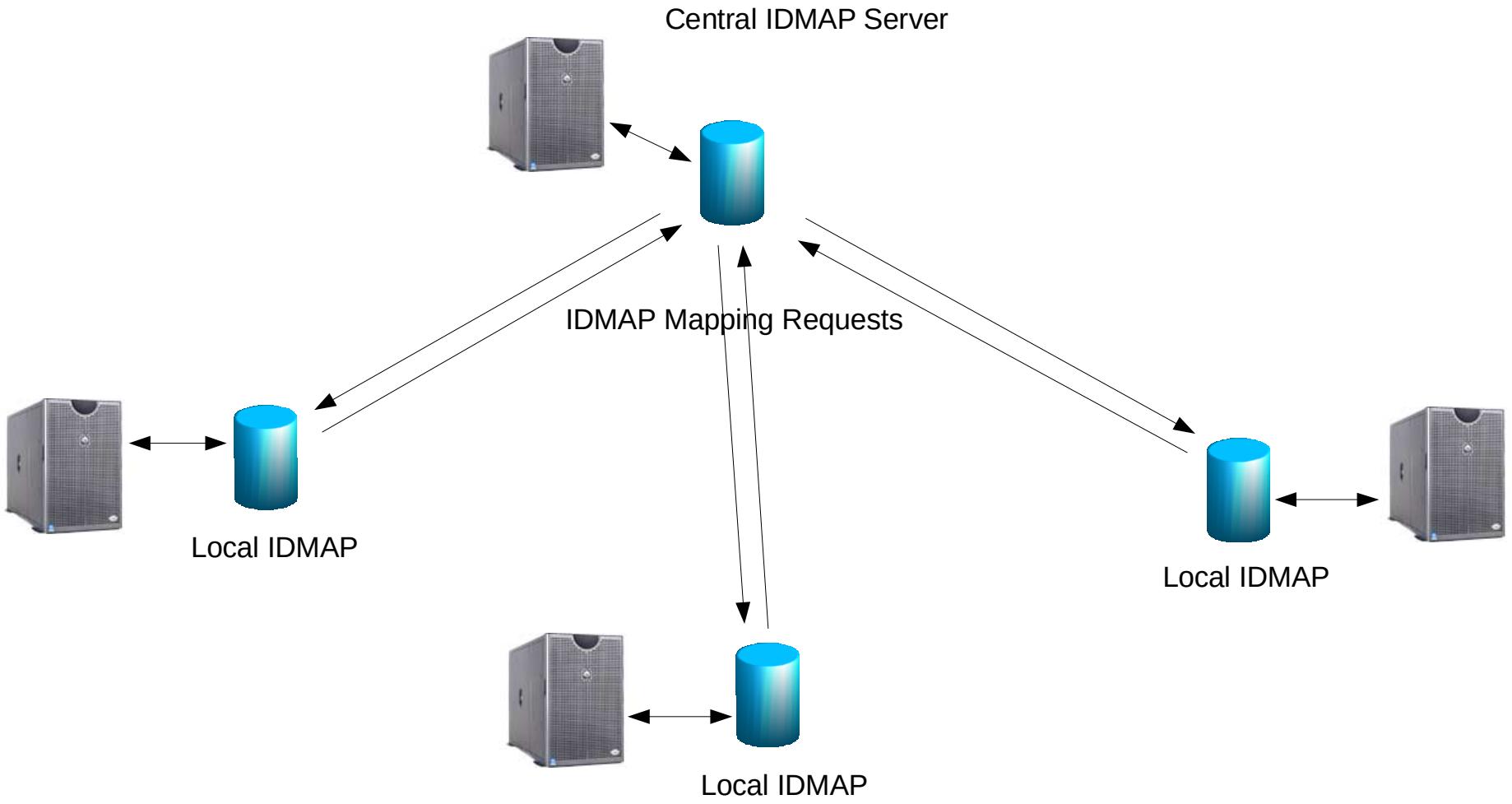


IDMAP

sID<->[u,g]ID MAPping

- Only map SIDs to UID/GIDs, nothing else
- It is a “persistent cache”
- SID<->[U,G]ID mapped when(if) needed

IDMAP with multiple servers



IDMAP with multiple servers

- UIDs,GIDs allocate randomly
- All kept consistent by a central server
- The central server handle all the mappings
- Peripheral servers keep a “permanent cache”

[U,G]ID Exhaustion

- SID space is a lot bigger than UID/GID space
- changing a mapping can be a security issue
- Changes will be an admin responsibility
- A notification mechanism based on sequence numbers will be implemented

SAM vs GUMS

- A brief history of the internal fork
  - Passdb
  - SAM
  - GUMS
- Dead paths
  - Multiple domain support
  - Multiple backends active at same time
- What we wanted:
  - The perfect SAM (accounts, privs, ecc..)
  - The perfect IDMAP
  - Winbind on PDC

How to Proceed

- Real needs:
  - A system that is good enough

Samba 3.0 Out!

- What will be into 3.0?
  - IDMAP
  - A possibly improved passdb
  - Winbind on PDC (?)