

## Super Administration

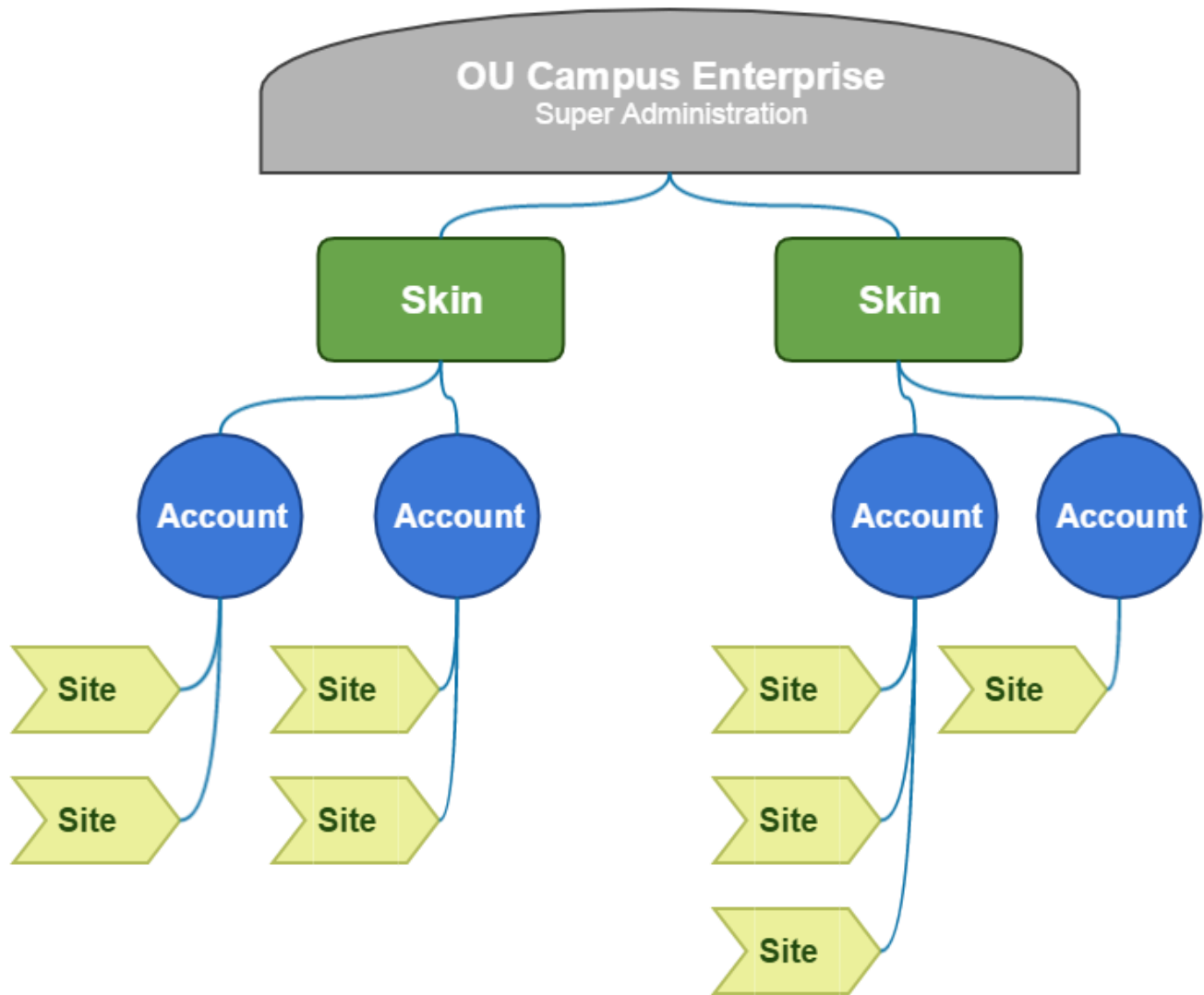
---

### Overview

OU Campus can be provided as a self-hosted installation, which is a local server maintained by the institution, or provided as SaaS (software as a service). The SuperAdmin interface provides administrators within said interface the ability to add, modify, and delete accounts, sites, and additional administrators, as well as access reports about users and groups. Level 5 administrators within Super Administration have the same ability with the exception of not being able to access the Administrators list view and the functionality available there.

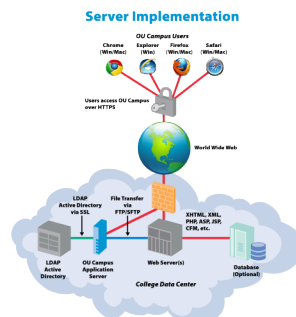
The various components of OU Campus can be deconstructed into the following hierarchy:

- OU Campus Super Admin: Can contain multiple skins. Administrators for the Super Administration interface can be added at this level.
  - Skins: Can contain multiple accounts, denoted by `/#skinname/` in the URL of the staging server.
    - Accounts: Can contain multiple sites, denoted by the string following `/#skinname/` in the URL.
      - Sites: Contain web content to be managed. Denoted by the string following the skin name and the account name in the URL (e.g. `/#skinname/accountname/sitename/`).



Users in the normal OU Campus interface are separate from administrators in the SuperAdmin interface. A level 10 administrator in the OU Campus interface will not necessarily have access to the SuperAdmin interface.

For more information, visit the [Login](#) page.



Within a single OU Campus skin, which Super Administration controls, there can be multiple accounts and within each account there are users and sites, each with distinct publish-to locations. Users are administrated at the account level in OU Campus; CAS and Shibboleth authentication are also enabled through the account level. LDAP can be set up for individual administrators and users.

For more information, visit the [Authentication Methods](#) page.

For self-hosted server installations a self-maintaining database is installed. The OU Campus database is internally configured, self-managed, and requires zero maintenance.

For more information, visit the [Database Abstraction](#) page.