

Authorisation Models

JF Perrin

“Direct use”: restrict access to content and services **based on user attributes:**

- Easy to implement (mod_oauth_openidc, mod_shib, ...)

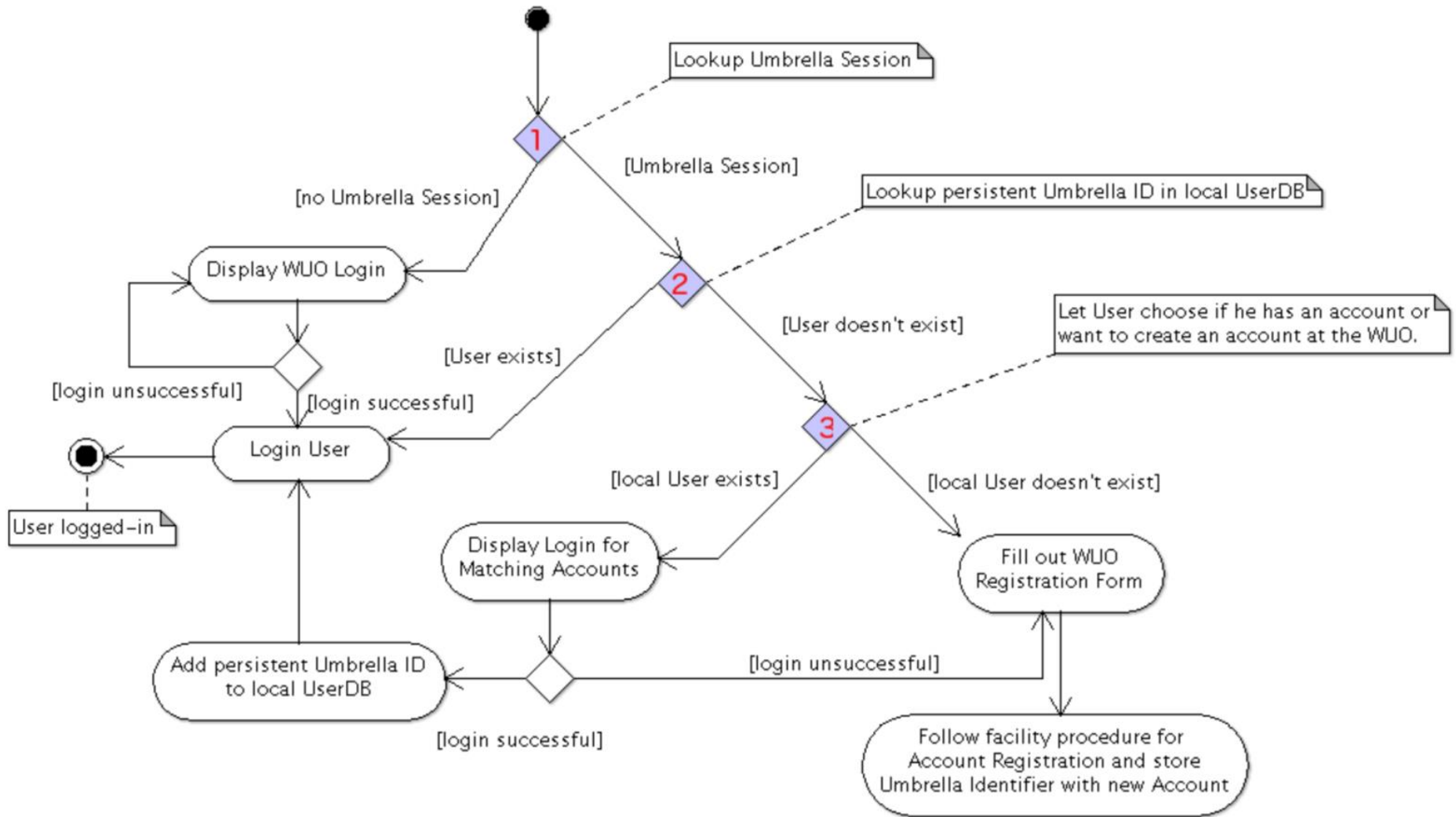
```
<Location /protected>  
  AuthType openid-connect  
  Require valid-user  
  Require claim family_name:Perrin  
</Location>
```

- Works for simple cases – you need to have access to users attributes that allow the authorisation decision.
- Typical use case: access to a phonebook, semi-public documents

- Up to now, only the EAAH (ID of the user).
- We will soon increase the list of possible attributes.
 - Name
 - Email
 - Affiliation
 - ORCID

- In your organisation authorisation mechanisms are in place and refer to a local ID
- Mapping the EAAH to a local ID
 - More complex workflow
 - Cumbersome to implement for all applications
 - Use of local SSO (Keycloak) to simplify the set up (Implement the mapping once for all the applications you want to expose).

Mapping workflow



Membership Management services

- VO specific **workflows** for onboarding members
- Registry for **user persistent Identifiers**
- Support for **R&S attributes** to maximize interoperability
- Use of **eduPersonEntitlement(s)** to express groups, roles and Service Entitlements
- Choice between **COmanage, HEXAA and Perun**

