



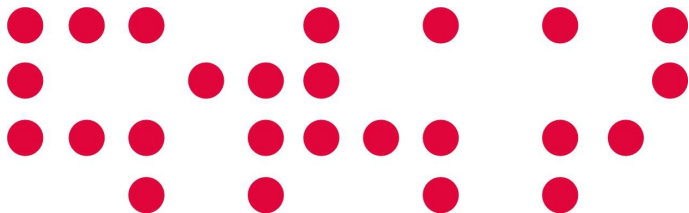
siLeDAP

Easing interactions with directories

Javi Masa, Cándido Rodríguez
{javier.masa,candido.rodriguez}@rediris.es

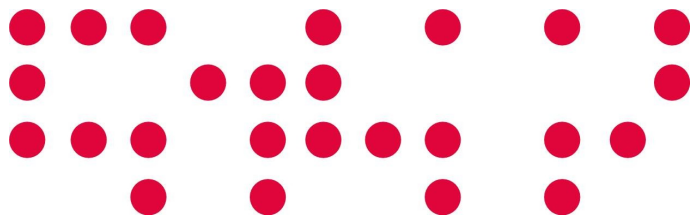
1. Goals

2. Description of the API
3. Demo: Navega2.0
4. COPA 2
5. Demo: Copa-Navega2.0
6. XLDAP
7. Any other question?



- **An advanced browser/administrator of LDAPs**
 - Being compatible with complex schemas
 - COPA
 - iris-*
 - Web 2.0 application
 - Using AJAX technology
- **API through web services**
 - Communication between Web 2.0 and LDAP
- **Wiki about LDAP**
 - Spanish documentation of LDAP technology
 - Learning LDAP is a bit difficult
 - Most common use cases.
- **<https://forja.rediris.es/projects/siledap>**

1. Goals
- 2. Description of the API**
3. Demo: Navega2.0
4. COPA 2
5. Demo: Copa-Navega2.0
6. XLDAP
7. Any other question?



- Defines a set of functions for LDAP interactions
- Those are exposed as web services
 - They can be called from a variety of platforms
 - They can be called from a variety of languages
- Not only atomic and basic operations
 - It's not necessary to implement in each application
 - COPA
- REST for requests and responses
 - Easier the integration between Ajax and LDAP
 - It's the most simple
- In the future, there will be available more protocols
 - XML-RPC, SOAP, JSON...

- Definition of the REST interface for requests:
 - It's a HTTP request
 - GET
 - `http://server/siledap/api/sw.php?par1=value1&par2=value2`
 - POST
 - Like data sent by forms
 - REST doesn't specify which services are available
 - We SHOULD read the documentation
 - Each web service defines how request should be made
 - We SHOULD read the documentation

- Example of a request

- Web service for performing searches in a LDAP
 - HTTP request, GET method
 - Mandatory parameters:
 - **dnbase**: value of the base DN in base 64 of the subtree.
 - Optional parameters:
 - **filter**: value in base 64 of the LDAP filter.
 - **scope**: “one” o “sub”
 - **selAttr**: list of attributes (joined with ,) in base 64.
- Example:

```
http://server/siLeDAP/API/searchLDAP.php?  
dnbase=ZGM9dW5pdixkYz1lcw==&  
scope=one&  
selAttr=ZGMsb3U=
```

- **REST format for responses**
 - It's a simple XML block
 - The elements that are contained in the XML message are different in each web service.
 - The documentation shows us the definition of the response

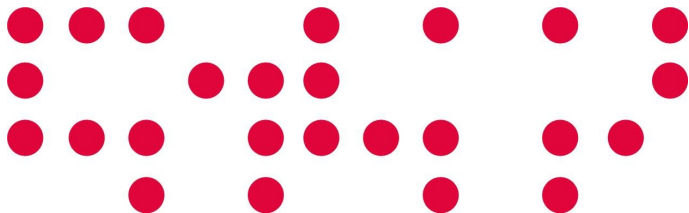
- Example of a response
 - For the last request:

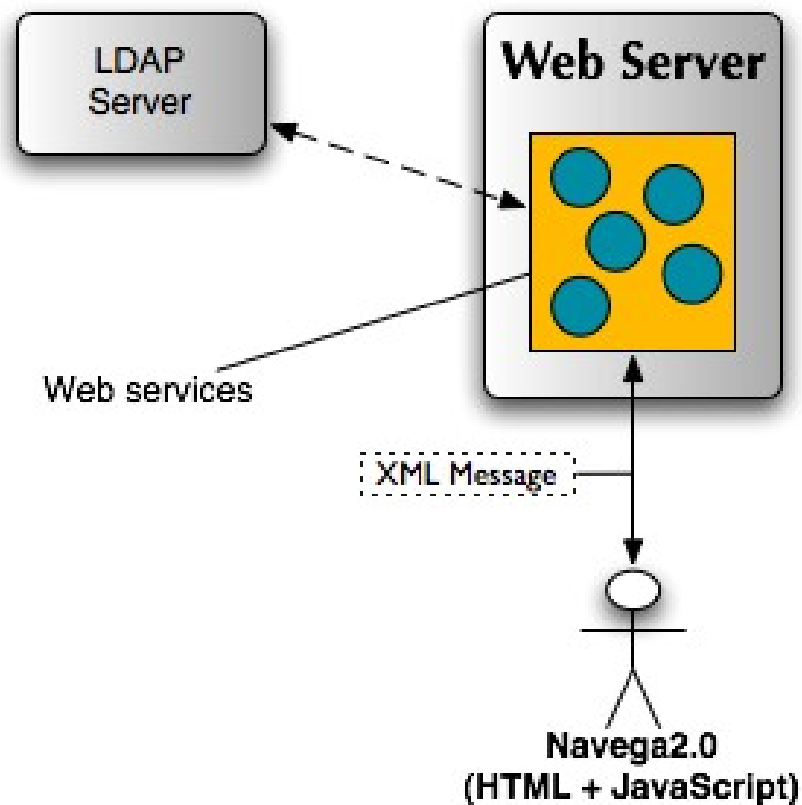
```
<?xml version='1.0' encoding='UTF-8'?>
<Entries>
  <Entry dn="dc=areas,dc=univ,dc=es">
    <Attribute name="dc">
      <AttributeValue value="areas"/>
    </Attribute>
    <Attribute name="ou">
      <AttributeValue value="areas"/>
    </Attribute>
  </Entry>
  <Entry dn="dc=catalogos,dc=univ,dc=es">
    <Attribute name="dc">
      <AttributeValue value="catalogos"/>
    </Attribute>
    <Attribute name="ou">
      <AttributeValue value="catalogos"/>
    </Attribute>
  </Entry>
</Entries>
```

- **Basic operations with LDAP**
 - Get an entry
 - By its DN
 - Performing searches
 - Base DN
 - Filter
 - Scope
 - Attributes
 - Add/Modify/Replace/Delete entries
 - Import/Export in LDIF
 - Queries to the schema

- **Helper operations with LDAP**
 - Get an image
 - By its DN
 - Specifying which attribute. Example: **jpegPhoto**
 - Get a certificate
 - By its DN
 - Specifying which attribute. Example: **userCertificate**

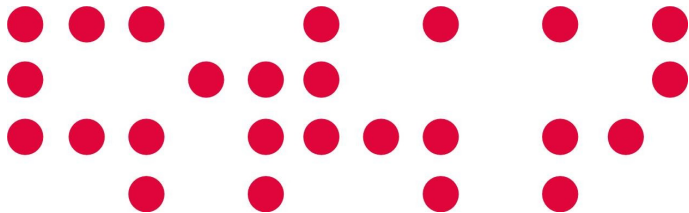
1. Goals
2. Description of the API
- 3. Demo: Navega2.0**
4. COPA 2
5. Demo: Copa-Navega2.0
6. XLDAP
7. Any other question?



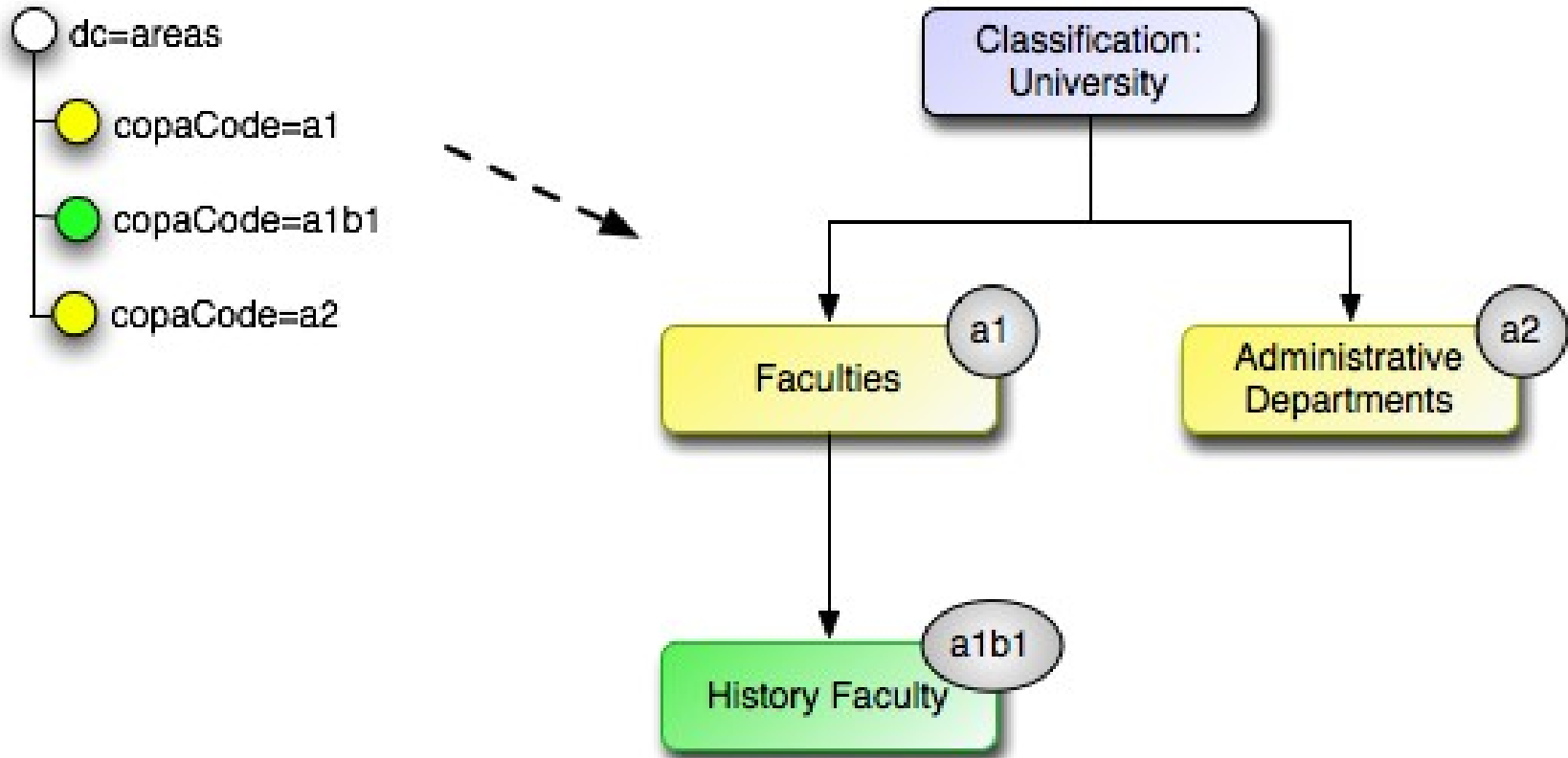


- Demo
 - LDAP Server of RedIRIS

1. Goals
2. Description of the API
3. Demo: Navega2.0
- 4. COPA 2**
5. Demo: Copa-Navega2.0
6. XLDAP
7. Any other question?

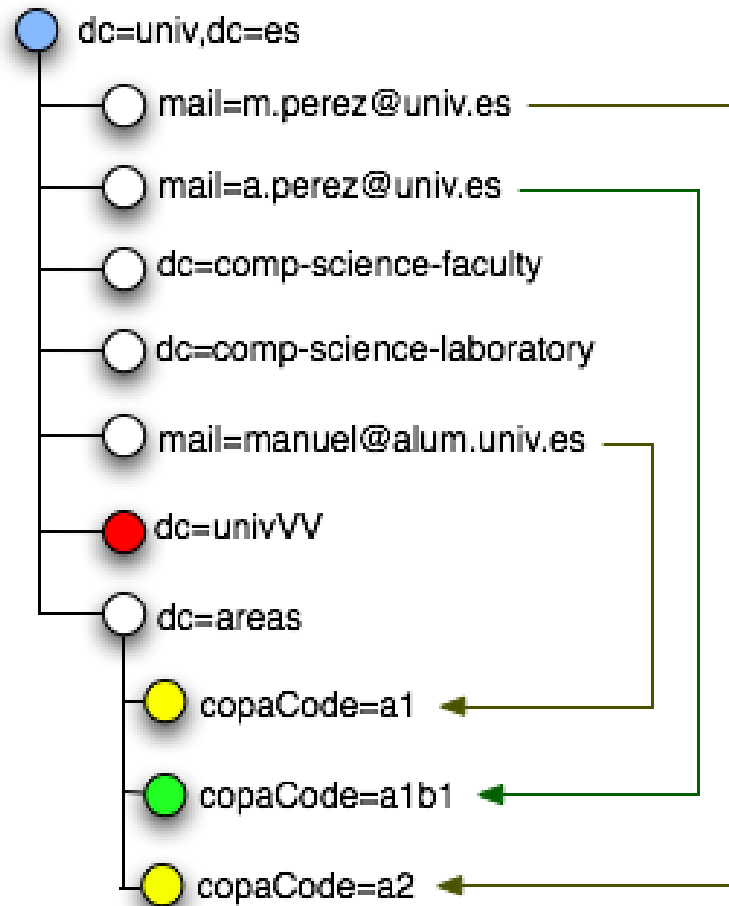


- COPA is a coding schema that allows effective searches in hierarchical structures.
- It's able to perform searches using metadata that is included in the pre-existing structure.
- Elements in COPA
 - Classifications
 - A set of areas in a tree structure
 - Alphanumeric codes (COPA code)
 - Relationships between different areas
 - Either different classifications

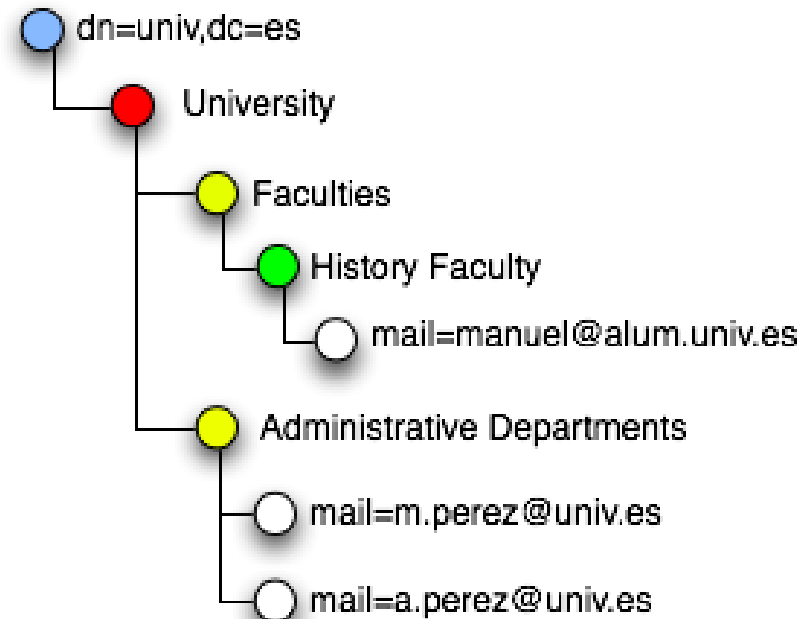


- COPA
 - Resources
 - Elements are associated to areas of different classifications
 - **Main goal:**
 - Create virtual views over a LDAP server

Plain LDAP



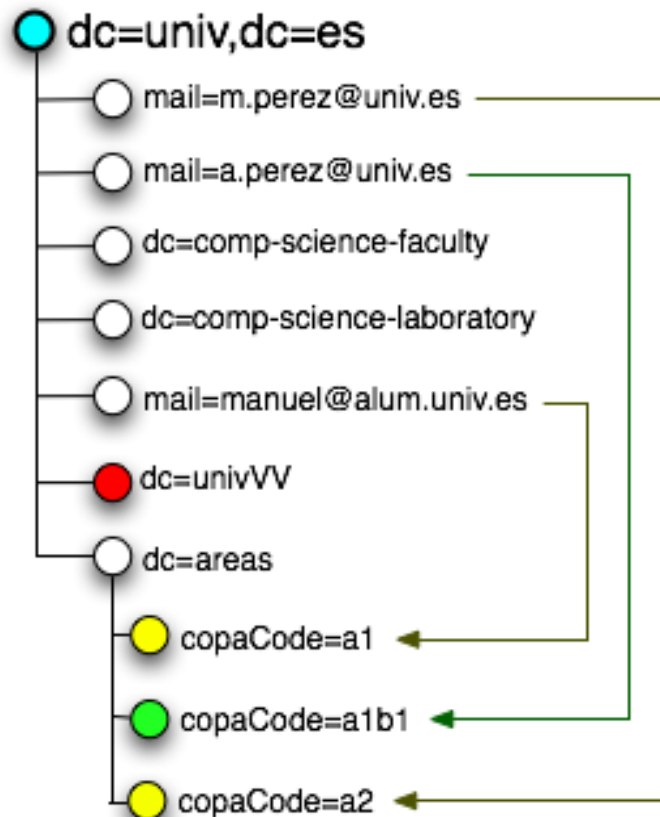
COPA-based Virtual View



- COPA 2

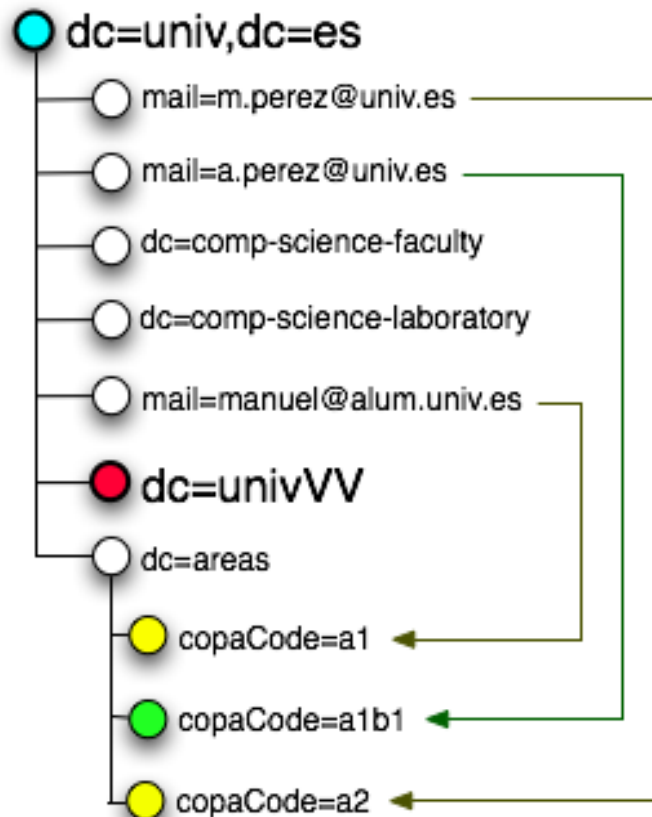
- Goal: applications understand COPA automatically.
- New object classes
 - copaVirtualViewNav
 - Defines which virtual views are available for navigate
 - copaVirtualViewSpec
 - Configuration of a virtual view
 - Where and how are areas defined
 - Which attribute is used for making a relationship between an entry and an area
 - DN are bad guys, URN are good
 - copaArea
 - Generic object for defining an area
 - copaResource
 - Generic object for making a relationship between an entry and an area

Plain LDAP



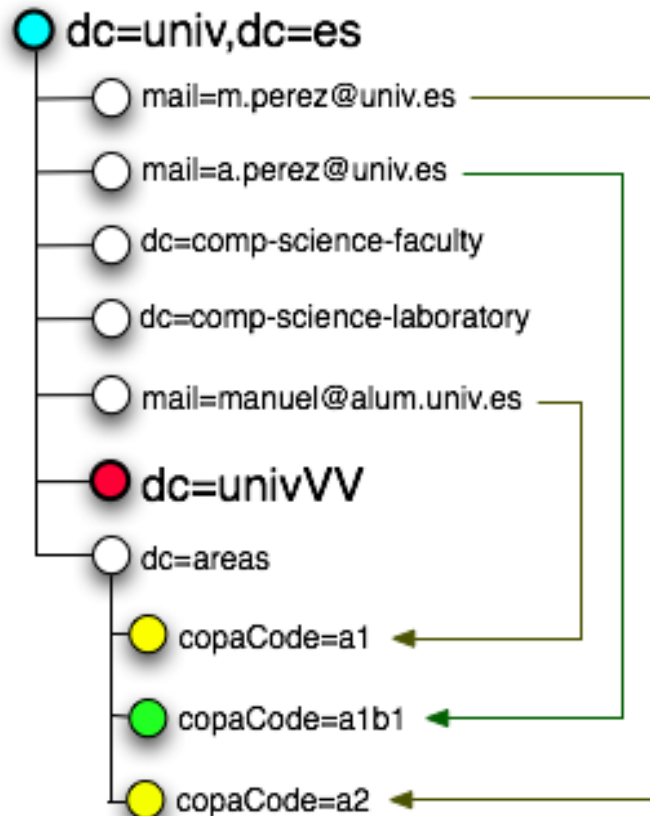
- **dc=univ,dc=es**
 - **objectClass**
 - organizationalUnit
 - dcObject
 - copaVirtualViewNav
 - **dc**
 - univ
 - **ou**
 - univ
 - **copaDefVvNav**
 - dc=univVV,dc=univ,dc=es

Plain LDAP



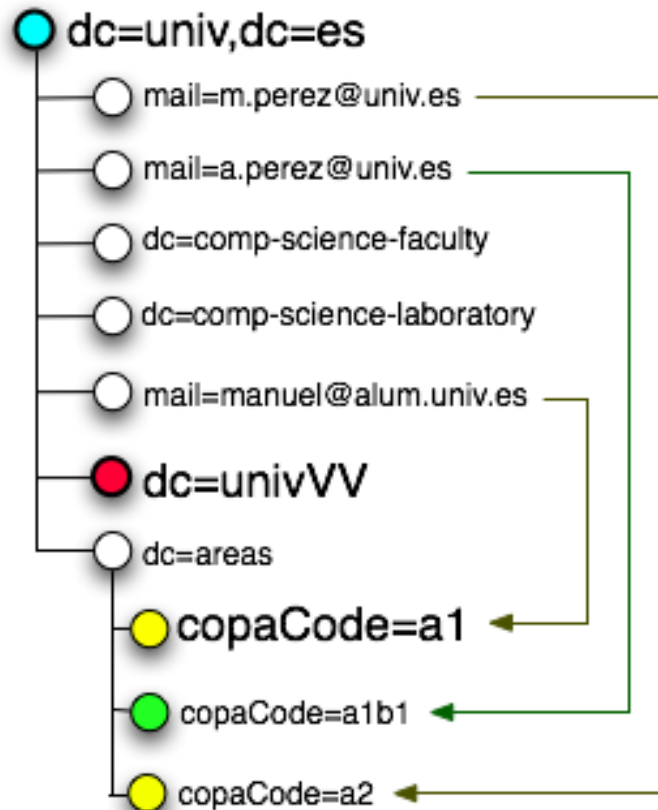
- `dc=univVV,dc=univ,dc=es`
 - **objectClass**
 - `dcObject`
 - `copaVirtualViewSpec`
 - **dc**
 - `univVV`
 - **copaName**
 - `University`
 - **description**
 - `An hierarchy of the university`
 - **copaPrefix**
 - `urn:mace:rediris.es:virtualview:univ:20061101-1.0`
 - **copaCodeAttr**
 - `copaCode`
 - **copaPrintAttr**
 - `copaName`

Plain LDAP



- **dc=univVV,dc=univ,dc=es**
 - **copaCodeResourceAttr**
 - copaCode
 - **copaClassifBase**
 - dc=areas,dc=univ,dc=es
 - **copaAreaObjectClassName**
 - copaArea

Plain LDAP

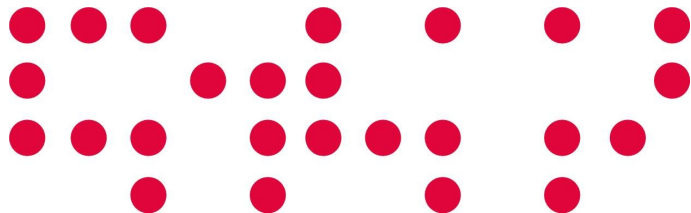


- `copaCode=a1,dc=areas,dc=univ,dc=es`
 - **objectClass**
 - `copaArea`
 - **copaName**
 - Faculties
 - **description**
 - List of faculties
 - **copaAreaCode**
 - `a1`

- COPA operations in a LDAP server (I)
 - Set the beginning of a COPA-based navigation
 - Adding the object class *copaVirtualViewNav*
 - Get a list of available virtual views
 - Which the default navigation is
 - Get a list of areas
 - DN of the used virtual view
 - Optional: COPA code of the parent area
 - Get info about an area
 - DN of the used virtual view
 - COPA code of the area

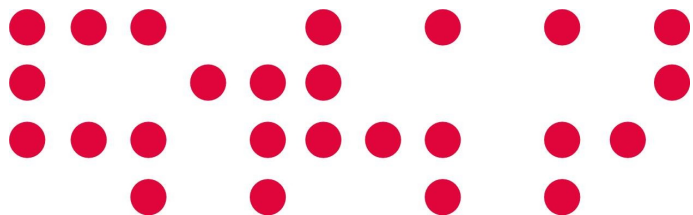
- COPA operations with a LDAP server (II)
 - Get a list of resources
 - Base DN of the search
 - DN of the used virtual view
 - COPA code of the area

1. Goals
2. Description of the API
3. Demo: Navega2.0
4. COPA 2
- 5. Demo: Copa-Navega2.0**
6. XLDAP
7. Any other question?



- Demo
 - LDAP server of a generic university

1. Goals
2. Description of the API
3. Demo: Navega2.0
4. COPA 2
5. Demo: Copa-Navega2.0
- 6. XLDAP**
7. Any other question?



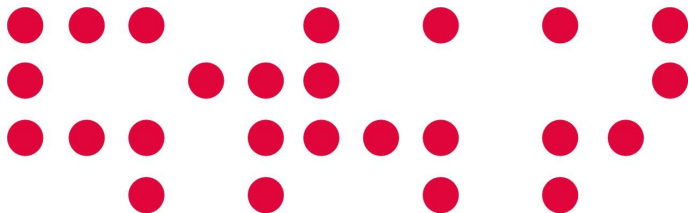
- Draft de IETF
 - Goal: to work with a LDAP server using XML messages
- Proposed protocol: XLDAP
 - Equivalent to LDAPv3
 - ASN.1-based operations converted to *Robust XML Encoding Rules* (RXER)
 - XML messages over TCP/IP and SOAP 1.1
 - The specification requires TCP/IP and, optionally, SOAP
- An example of a XLDAP message...

```
<xed:LDAPMessage
  xmlns:xed="urn:ietf:params:xml:ns:xed"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xldap="urn:ietf:params:xml:ns:xed-uldap"
  xsi:type="xldap:LDAPMessage">
  <messageID> 0 </messageID>
  <protocolOp>
  <searchRequest>
  <baseObject> <!-- DistinguishedName -->
  <!-- Equivalent LDAPDN is dc=example,dc=com -->
  <item> <!-- RelativeDistinguishedName -->
  <item> <!-- AttributeTypeAndDistinguishedValue -->
  <type> 0.9.2342.19200300.100.1.25 <!-- dc --> </type>
  <value>com</value>
  </item>
  </item>
  <item>
  <item>
  <type> 0.9.2342.19200300.100.1.25 <!-- dc --> </type>
  <value>example</value>
  </item>
  </item>
  </baseObject>
  <scope>wholeSubtree</scope>
  <derefAliases>derefInSearching</derefAliases>
  <sizeLimit>100</sizeLimit>
  <timeLimit>5</timeLimit>
  <typesOnly>>false</typesOnly>
  <filter>
  <and>
  <filter> <!-- objectClass = person -->
```

```
</attributeDesc>
  <assertionValue> 2.5.6.6 <!-- person -->
</assertionValue>
  </equalityMatch>
</filter>
<filter> <!-- surname = Smith -->
  <equalityMatch>
  <attributeDesc>
  <type> 2.5.4.4 <!-- surname --> </type>
  </attributeDesc>
  <assertionValue>Smith</assertionValue>
  </equalityMatch>
</filter>
</and>
</filter>
  <attributes> <!-- all attributes --> </attributes>
</searchRequest>
</protocolOp>
</xed:LDAPMessage>
```

- **What does siLeDAP think about XLDAP?**
 - Providing a web service which implements the SOAP interface of XLDAP
 - When there'd be any XLDAP client
 - XLDAP is not the solution for making interoperations with LDAP easier.
 - Also, it's too complex for integrating with AJAX

1. Goals
2. Description of the API
3. Demo: Navega2.0
4. COPA 2
5. Demo: Copa-Navega2.0
6. XLDAP
7. Any other question?





Thank you for your attention

Any other question?