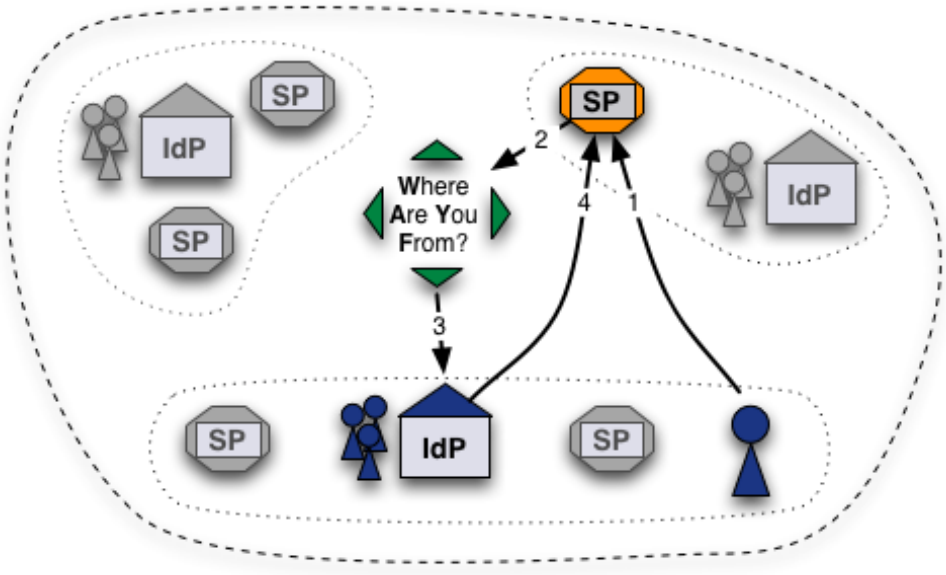
Discovery Service Options

SWITCH

SWITCHaai Team
aai@switch.ch

The classic way: One WAYF per Federation



WAYF achieves high availability through redundancy and IP Anycast.

Alternatives to Central WAYF

- Direct Login URLs
- SWITCH Embedded WAYF

SWITCH WAYF

- Shibboleth Embedded Discovery Service

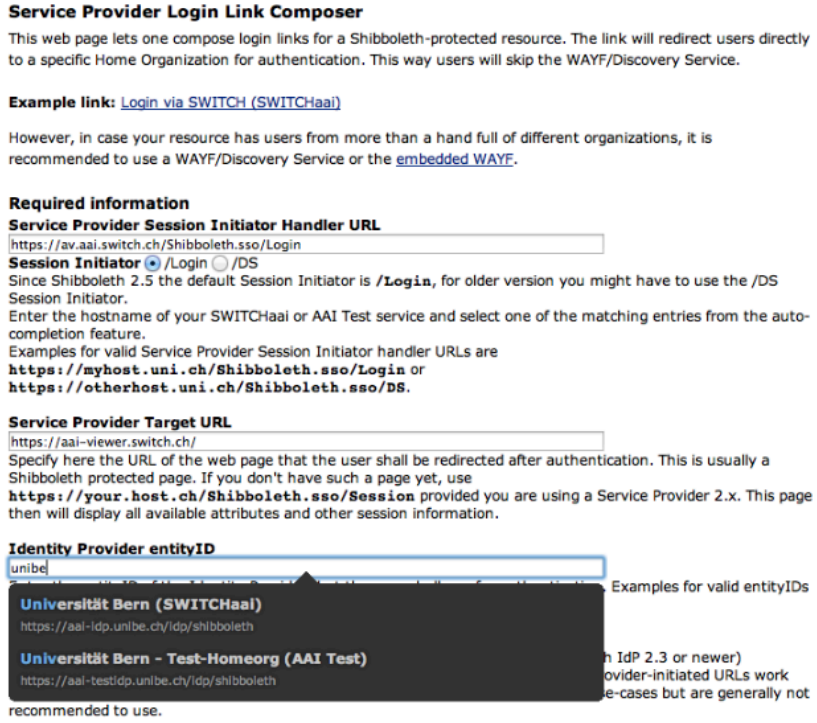
Shibboleth EDS

Solution 1: Direct Login URLs

- A separate login link for a specific IdP
- 1 click direct redirect to IdP without going via WAYF
- Useful when only users of few IdPs use resource

Login links:
Login via SWITCH (SWITCHaai)
Login via Stockholm University (Interfederation)
Login via University of Gothenburg (Interfederation)

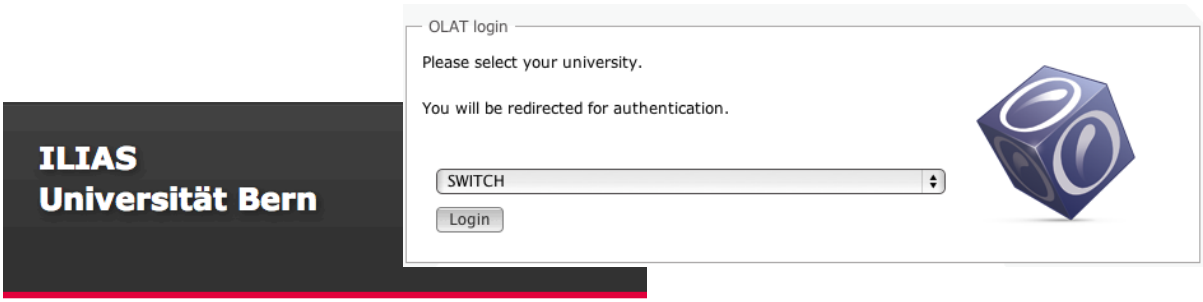
Composing Login URLs

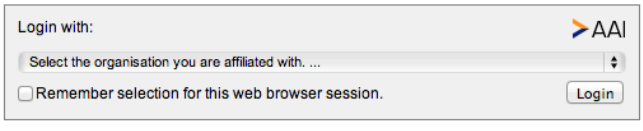


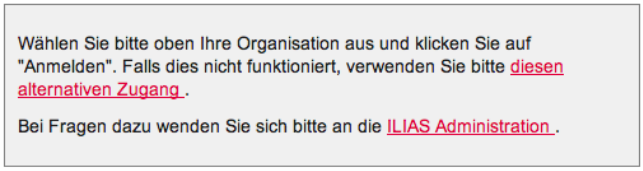
https://www.switch.ch/aai/support/serviceproviders/sp-compose-login-url.html

Solution 2: Embedded WAYF

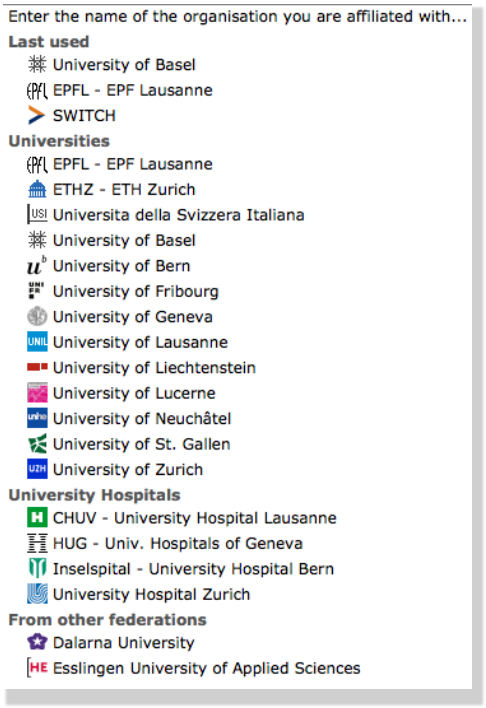
SWITCH WAYF





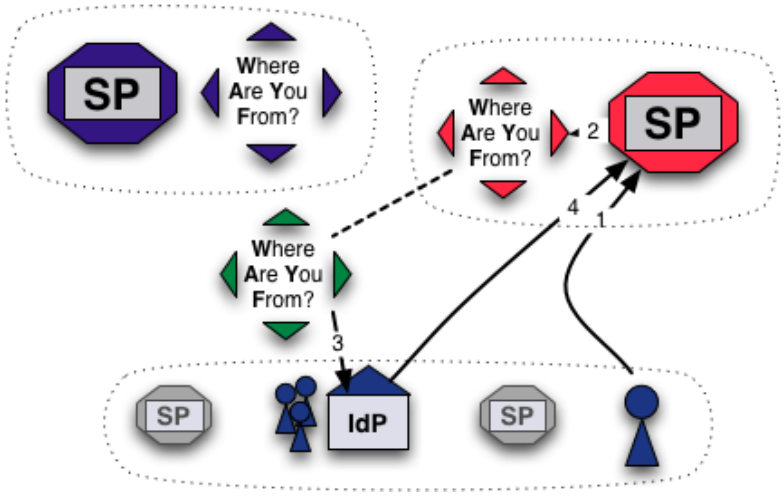


Embedded WAYF



Embedded WAYF

- Embed WAYF on Web Application
- customize look and feel
- still transparently uses central WAYF



Information and Configuration

More information about the Embedded WAYF:

https://www.switch.ch/aai/support/serviceproviders/sp-embedded-wayf.html

Generate the Embedded WAYF code for your SP:

https://rr.aai.switch.ch/gen_embedding_code.php

Configuration

Configuration Example of Embedded WAYF

```
// Example of how to add Identity Provider from other federations
var wayf_additional_idps = [
        {name:"Esslingen University of Applied Sciences",
        entityID:"https://idp.hs-esslingen.de/idp/shibboleth",
        logoURL:"https://www2.hs-esslingen.de/favicon.ico"
        },
        {name:"Dalarna University",
        entityID:"https://login.du.se/idp/shibboleth",
        logoURL:"https://login.du.se/duse-logo-16x16.png"
        }
];
```

Configuration (2)

SWITCH WAYF

Configuration Example of Embedded WAYF

```
// EntityIDs of Identity Provider that should not be shown at all
// [Optional, commented out by default]

var wayf_hide_idps = new Array ("https://idemfero.units.it/idp/shibboleth",
"https://idp.it.su.se/idp/shibboleth");

// Categories of Identity Provider that should not be shown
// Possible values
   are:"university","uas","hospital","library","vho","others","all"

var wayf_hide_categories =  new Array("library","vho","others","hospital");
```


Enable JSON Discovery feed to use local metadata of SP

Shibboleth SP

In shibboleth2.xml:

```
  <Sessions lifetime="28800"
                timeout="3600"
                relayState="ss:mem"
                checkAddress="false"
                consistentAddress="true"
                handlerSSL="true"
                cookieProps="https">
...
  <!-- JSON feed of discovery information. -->
       <Handler type="DiscoveryFeed" Location="/DiscoFeed"/>
  </Sessions>
```

JSON Discovery feed example

Shibboleth SP

```
[ { "entityID": "https://tlogin.usi.ch/idp/shibboleth",
 "DisplayNames": [
 { "value": "Universita della Svizzera Italiana", "lang":
"en" },
 { "value": "Universita della Svizzera Italiana", "lang":
"it" }
 ]
},
{ "entityID": "https://dtaai.unil.ch/test/idp/shibboleth",
"DisplayNames": [
 { "value": "Université de Lausanne Test", "lang": "en" },
 { "value": "Université de Lausanne Test", "lang": "fr" }
 ],
]
```


Configuration (3)

SWITCH WAYF

Configuration Example of Embedded WAYF

```
// Whether to load Identity Providers from the Discovery Feed provided by
// the Service Provider.
// IdPs that are not listed in the Discovery Feed and that the SP therefore is
// not able to accept assertions from, are hidden by the Embedded WAYF
// IdPs that are in the Discovery Feed but are unknown to the SWITCHwayf
// are added to the wayf_additional_idps.
// The list wayf_additional_idps will be sorted alphabetically
// The SP must have configured the discovery feed handler that generates a
// JSON object. Otherwise it won't generate the JSON data containing the IdPs.
// [Optional, default:false]

var wayf_use_disco_feed = true;
```

Solution 3:
Embedded Discovery Service

Shibboleth EDS

- Requires the Discovery Feed provided by the SP
- Embed the DS directly into the service
- Search-as-you-type or select from list
- JavaScript, CSS and HTML only
- developed and maintained by the Shibboleth team
- download from

https://shibboleth.net/downloads/embedded-discovery-service/latest/

- Documentation can be found at:

https://wiki.shibboleth.net/confluence/display/EDS10/Embedded+Discovery+Service

Embedded Discovery Service

Shibboleth EDS

MetadataFilter Example

Shibboleth SP

In shibboleth2.xml:

```
<MetadataProvider type="XML" .....>

    <MetadataFilter type="Whitelist">
       <Include>https://testidp.unifr.ch/idp/shibboleth</Include>
       <Include>https://aai-demo-idp.switch.ch/idp/shibboleth</Include>
       <Include>https://aai-testidp.unibe.ch/idp/shibboleth</Include>
    </MetadataFilter>

</MetadataProvider>
```


Embedded WAYF vs Embedded DS

Properties	Login Link	Embedded WAYF (SWITCH)	EDS (Shibboleth)
Independent from central server	✓		✓
Display only “valid” IdPs for SP		(✓)	✓
Search as you type feature		✓	✓
Show Home Org Logo	(✓)	✓	✓
Very easy deployment	✓	✓	✓
Can be used with old SPs (<2.4)	✓	(✓)	
Categories supported	(✓)	✓	
Uses cached recent IdP selection across different services		✓	

When to use what ?

Numbers of IdPs	Login Link(s)	Embedded WAYF SWITCH	EDS Shibboleth
1 - 5	✓	✓	✓
1 - 50		✓	✓
1 - 500		✓	✓

To mention: Disco Juice

- Very comprehensive Discovery Service
- Well suited for services with users from many IdPs
- Search-as-you-type only
- Uses Geo IP and metric to guess user's IdP
- Based on PHP and JS

http://discojuice.org/