SWORD 2.0

Supporting Full Deposit Lifecycles

Richard Jones
Cottage Labs
richard@cottagelabs.com
http://cottagelabs.com
What is SWORD?

SWORD is a:
Simple
Web service
Offering
Repository
Deposit

During the development of SWORD it was decided to keep the protocol as simple and lightweight as possible.
What is SWORD?

SWORD is a:
Simple
Web service
Offering
Repository
Deposit

A software interface that is accessible using http
What is SWORD?

SWORD is a:
Simple
Web service
Offering
Repository
Deposit
What is SWORD?

SWORD is a:
- Simple
- Web service
- Offering
- Repository
- Deposit

Agnostic:
- DSpace
- EPrints
- Fedora
- Intralibrary
- Zentity
- ???
What is SWORD?

SWORD is a:
Simple
Web service
Offering
Repository
Deposit

Putting things INTO a repository
What is the purpose of SWORD?

The purpose of SWORD is to:
Provide an *interoperable interface* that allows *deposits* to be easily made *into* repositories.
The SWORD project

Out of scope:
Packaging
Metadata
Authentication
Authorization
SWORD use cases

Deposit from a desktop client

e.g. DepositMO
SWORD use cases

Deposit to multiple repositories

- Institutional repository
- Subject-based repository
- Funder repository

e.g. OA-RJ
SWORD use cases

Deposit by lab equipment
SWORD use cases

Repository to repository deposit

Institutional repository → <sword /> → National repository
SWORD use cases

Publishing system / publisher to repository deposit

Publishing system (e.g. OJS) → <sword /> → Long term storage repository

e.g. OJS client
Bibapp client
SWORD use cases

Research Information System to repository deposit

- Research Information System
- <sword />
- Long term storage repository

E.g. Bibapp client
RePosit project
How does SWORD work?
HTTP basics

There are different methods / verbs that we use when interacting with the web:

**GET**

  - Requests a resource

**POST**

  - Create a new resource

**PUT**

  - Update/Replace an existing resource

**DELETE**

  - Delete a resource
SWORD foundations

SWORD is a profile of an existing standard Atom Publishing Protocol (AtomPub)

AtomPub is for publishing content to the web

AtomPub is designed to be extended

AtomPub is used by other systems:
GData (used by Google Docs etc), CMIS
SWORD foundations

SWORD is a profile of an existing standard Atom Publishing Protocol (AtomPub)

What does this mean?
SWORD isn’t re-inventing the wheel
AtomPub clients / APIs can be re-used
Two step process

1) Ask the repository’s SWORD interface to describe itself

2) Use that information to prepare and make a deposit
Introducing: The Service Document

Ask the repository’s SWORD interface to describe itself

This is done using a Service Document

A service document describes the contract between a user and a repository: what you must provide and what the repository will do in return
How to request a Service Document

Request a service document from the repository

**HTTP GET** a service document

- Lists the Collections you can deposit into
- Tells you what formats are supported for deposit
- Tells you what will happen to your content
The second step

We have retrieved the Service Document
We know where we can deposit
We know what we can deposit
We know what will happen to our deposit
We have created a deposit package

We perform the deposit...
Performing a deposit

To perform a deposit:

**HTTP POST** a file to the deposit URL

- POST /geography-collection HTTP/1.1
- Content-Type: application/zip
- Content-Disposition: filename=MyItem.zip
- Packaging:
  [http://purl.org/net/sword/packages/default](http://purl.org/net/sword/packages/default)
Introducing: The Deposit Receipt
Is there anything else we can do?

YES!

Where SWORD 1.3 leaves off SWORD 2.0 begins...

The deposit lifecycle is more complicated than fire-and-forget...
Get a package of the item's entire content in our favourite format

Get a package of the item's entire content in our favourite format.
Add more content to the item by POSTing more files

<entry xmlns="http://www.w3.org/2005/Atom" xmlns:sword="http://purl.org/net/sword/">
    <title>My Deposit</title>
    <id>info:something:1</id>
    <updated>2008-08-18T14:27:08Z</updated>
    <summary type="text">A summary</summary>
    <generator uri="http://myrepository.ac.uk/engine" version="1.0"/>
    <content type="application/zip"
        src="http://myrepository.ac.uk/geography-collection/deposit1.zip"/>
    <link rel="edit"
        href="http://www.myrepository.ac.uk/geography-collection/my_deposit.atom" />
    <link rel="edit-media"
        href="http://www.myrepository.ac.uk/geography-collection/my_deposit" />
    <link rel="http://purl.net/sword/terms/statement"
        href="http://www.myrepository.ac.uk/geography-collection/my_deposit.rdf" />
</entry>
PUT new content over the old, or
DELETE all the content
Ask the repository to describe the item, and the state it is in
A Banking Analogy

You Deposit your money

You get a Receipt

Your Statement tells you what happened to your money since
Introducing: The SWORD Statement

It tells you what files were produced when the repository unpackaged your deposit

It tells you what new files have been added to your deposit (e.g. via format migration, mediated deposit, etc)

It tells you where to get the package(s) you originally uploaded

It tells you where in the workflow your item is (or any other state that it might be in). For example, “under review” or “deleted”
Can be an OAI-ORE Resource Map

```xml
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:ore="http://www.openarchives.org/ore/terms/"
         xmlns:sword="http://purl.org/net/sword">

   <rdf:Description rdf:about="http://localhost/edit-uri/43/d0">
     <ore:describes rdf:resource="http://localhost/agg-uri/43/d0"/>
   </rdf:Description>

   <rdf:Description rdf:about="http://localhost/agg-uri/43/d0">
     <ore:isDescribedBy rdf:resource="http://localhost/edit-uri/43/d0"/>
     <ore:aggregates rdf:resource="http://localhost/part-uri/43/d0/example.zip"/>
     <sword:originalDeposit rdf:resource="http://localhost/part-uri/43/d0/example.zip"/>
   </rdf:Description>

   <rdf:Description rdf:about="http://localhost/part-uri/43/d0/example.zip">
     <sword:depositedBy>sword</sword:depositedBy>
   </rdf:Description>

   <rdf:Description rdf:about="http://purl.org/net/sword/state/archived">
     <sword:stateDescription>
     The work has passed through review and is now in the archive
     </sword:stateDescription>
   </rdf:Description>

</rdf:RDF>
```
Tells me where to get my original package
Tells me what state my item is in

The work has passed through review and is now in the archive
User friendly?

Interaction with SWORD is via XML documents

XML is good for machine to machine interfaces

Do users want to read / write XML?
User friendly?
The answer?

SWORD clients
An analogy

Using the web:

Do we read and write HTML?

No!

We use a web browser.
SWORD clients

Three types of SWORD client:

- **Machine to machine custom client**
  - E.g. custom code to deposit from lab equipment

- **General client**
  - Deposit any item type to any repository

- **Specific client**
  - Tailored for an individual use (e.g. e-thesis deposit to repository X, journal article deposit to repository Y)
Online demo client

SWORD Servlet Client - select a service document

This client can access services that conform to the SWORD 1.3 profile.

Select a service document, or enter another URL:

URL: http://client.swordapp.org/client/servicedocument

Other: 

Username: sdl@aber.ac.uk Password: ********

On behalf of: 

Get Service Document

SWORD is a JISC funded project managed by UKOLN. For further information about SWORD, please visit: www.swordapp.org.
Desktop demo client
Another example

The Facebook SWORD deposit tool

http://apps.facebook.com/swordapp/
Facebook client

SWORDAPP – The Repository Deposit Tool

Welcome back to SWORDAPP Stuart!

Home  Deposit an item  See my items  See items by my friends  About SWORDAPP

Deposit an item – Step 2: Select collection

- Server details: Connection OK!  (show details)

- Please choose from one of the following collections to make your deposit:
  - Workspace: DSpace SWORD 1.3 Demo
  - Collection: Data sets (A collection for depositing data sets)
    - Deposit into this collection
    - See full collection details...
  - Collection: Research materials (A collection for depositing research materials)
    - Deposit into this collection
    - See full collection details...
  - Collection: Teaching materials (A collection for depositing teaching materials)
    - Deposit into this collection
    - See full collection details...
  - Collection: Workflow (Collection with workflow)
    - Deposit into this collection
    - See full collection details...
Facebook client

Deposit an item - Step 3: Item details

Deposit to: http://dspace.swordapp.org/sword/deposit/123456789/4
Collection title: Research materials
Abstract: A collection for depositing research materials

<table>
<thead>
<tr>
<th>Type of item:</th>
<th>Journal article</th>
</tr>
</thead>
<tbody>
<tr>
<td>Has the item been peer reviewed:</td>
<td>Yes</td>
</tr>
<tr>
<td>Title:</td>
<td></td>
</tr>
<tr>
<td>Abstract:</td>
<td></td>
</tr>
<tr>
<td>1st author first name:</td>
<td></td>
</tr>
<tr>
<td>1st author surname:</td>
<td></td>
</tr>
<tr>
<td>2nd author first name (optional):</td>
<td></td>
</tr>
<tr>
<td>2nd author surname (optional):</td>
<td></td>
</tr>
<tr>
<td>3rd author first name (optional):</td>
<td></td>
</tr>
<tr>
<td>3rd author surname (optional):</td>
<td></td>
</tr>
<tr>
<td>Bibliographic citation (optional):</td>
<td></td>
</tr>
<tr>
<td>Existing URL for item (optional):</td>
<td></td>
</tr>
</tbody>
</table>

Next >
Facebook client

The application performs the packaging (metadata + file)
Welcome back to SWORDAPP Stuart!

Deposit an item – item deposited!

You have successfully deposited 'My title' whose URL is now http://dspace.swordapp.org/jspui/handle/123456789/652
Deposit from within Microsoft Word

Article Authoring Add-in for Word 2007

DepositMO project (JISCdepo)

.docx files are actually ‘packages’

• Zip file containing XML files
<table>
<thead>
<tr>
<th>Clients</th>
<th></th>
<th>Servers</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PHP (+ Facebook)</td>
<td></td>
<td>DSpace</td>
<td></td>
</tr>
<tr>
<td>Java</td>
<td></td>
<td>EPrints</td>
<td></td>
</tr>
<tr>
<td>Python</td>
<td></td>
<td>Fedora</td>
<td></td>
</tr>
<tr>
<td>Ruby (+ Bibapp)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SWORD 2.0 Timeline

November – February
*Consultation and development of a draft profile and internet drafts*

March – May
*Client and Server Development*
Credits

Original slides by Stuart Lewis stuart@swordapp.org
Additional SWORD 2.0 material by Richard Jones richard@cottagelabs.com
The SWORD project http://swordapp.org/
Funded by JISC http://www.jisc.ac.uk/
Licence Creative commons