Slide 1 & 2 Technical issues

There are a wide variety of technical issues related to starting up an IR. I'm not a technical expert, so I'm going to cover most of these in a fairly superficial way to give you an idea of the types of issues you need to be prepared to think about. I'll cover each of these issues to some degree in this session

Slide 3 Technical expertise

Presumably you are all at institutions where you have a Systems Dept or other in-house expertise, can work with a computing center, or can hire technicians to bring up and maintain the system.

These are some of the broad technical skills you need to make sure you have in place or can get ready access to. The specifics will vary depending on the software you choose.

- You will need a knowledge of operating systems and servers **(continued ...)**

- You will need to have someone who understands the construction and operation of databases. Most likely, whatever software you choose will have certain reporting and administrative functions built in. Chances are also good that the standard functions will not be enough to meet all of your needs. You will want to extract different kinds of information,

- you will need to work around bugs and make things work smoothly so someone needs to be able to troubleshoot – and do it quickly. With an institutional repository, you want to be very responsive to any problems and get them resolved quickly. As a new service, a key aspect of your ability to market it will be its reliability.

Slide 4 Technical expertise

- Since the software for IRs is new and often open source, there are frequent updates. If you do any local customization, those local changes will need to be tracked and made to operate with updates. There will also be bug fixes and patches to install – and you'll need to make sure all the pieces function. This is nothing new to anyone who has used any software package

- You'll need to have people who can pull together pieces from a variety of sources and make them work together, especially if you're using open source software

- You'll want to make sure that your technical people have a good understanding of existing and emerging standards. The strength of such IRs is their stability and the discoverability of the metadata and the content. I'll get a bit more into this in a moment.

Slide 5 Hardware and software

There are a number of options available to you, things you will need to consider relating to the hardware and software.

- Do you want to go open source (meaning the software is freely available for download and installation). If it's open source, is there a strong user community that you can turn to for technical advice? Is there a community that is committed to improving the underlying code and sharing their updates and fixes, formally or informally?
- If you purchase or license software, what is the technical support like? Again, is there a user community you'll be able to tap into and get help from – because no piece of software I've ever worked with has ever worked 100% the way it's supposed to 100% of the time.
- Are you going to mount the IR on your own servers or will you contract with a vendor to do it for you? Either way, you need to be sure of having quick turnaround on resolving problems
- Whether you do this yourselves or contract it out, you'll want to know that the equipment is reliable and up to the task.
- And since you're capturing and storing other people's work, you'll want to be sure that the backup mechanisms are robust.

Slide 6 Software review

There are many software options available to you, open source and purchased or licensed. Some of these are open source and others are not. There are beginning to be some tools for helping prospective users compare the features and the challenges of the different options. **I've referred you in the supplemental material to some tools for evaluation.**

The software you choose is going to depend on many things.

Slide 7 Software requirements

As we started to evaluate software options at Oregon, we looked at a number of factors. These are not all absolute requirements but are the kind of things that you may at least want to consider.

In addition to basing this on our own experience, I have also consulted the new publication entitled "The Institutional Repository" written by (**Continued...)**

Richard Jones, Andrew, and John MacColl of the UK – an excellent work providing an overview of all aspects of setting up an IR just published by Chandos

- Permit the easy creation, use, and administration of digital objects distributed over the Internet
- I'll get more into some specifics of this when I talk about digital preservation
- Facilitate the creation of collections of materials in different disciplines or categories –
    - one of the things about an IR is that you'll want to be able to group together materials from the same department or program or course and you'll also want to be able to delineate and link items to more than collection or grouping

Slide 8 Software requirements

- Support any type of file – not just text-based files – the software should be able to handle a wide variety; in the policy section we'll get into whether you should accept all types of files, even if the software permits it

- Carry out searches based on standard metadata – is the metadata mapped to a standard set of data elements such as Dublin Core

- Flexible metadata capture, edit, and display – things like the ability to edit metadata easily and globally across fields; ability to key metadata, set up defaults on a collection-by-collection basis and override those defaults; ability to modify the elements that are searched or displayed on a collection-by-collection basis; ability to change labels for different metadata elements

Slide 9 Software requirements

- Plug into your local authentication system, such as LDAP (lightweight directory access protocol) that exists on many campuses – whether that be by signing on or based on a certificate

- Be constructed using components and technologies that were standard and non-proprietary
    - Things like supporting web service protocols, such as http

- Easily integratable – needs to fit well in to existing infrastructures and designs – desirable to be able to embed components of the IR into other existing systems

Slide 10 Software requirements

- Customizable user interface – as an institutional repository you want to be able to put your institution's brand on it (whether that's the library or the entire institution)

- Modular – so that if you do any local customization retaining those customized elements will not require too much effort when you do upgrades

- Flexible system administration – maintain the application quickly and easily. Ability to manage authorizations in groupings, make global changes to a group of users;

- Granular authorizations – different users are allowed to do different things. Ideally this should be flexible enough so that you can determine the combination of features that different users can do.

Slide 11 Software requirements

- Scalable – Can support the growth of the archive that you hope to have
- Manage licenses and permissions –
    - At least four parties must be tracked:
        - the owner of the content,
        - the submitter of the content (which is very definitely not the same person as the owner of the content all the time),
        - the institution,
        - and the user of the content –
    - desirable to have different licenses for different parties and to have a system that can manage the use of the content according to the terms and conditions of the different licenses
- Recoverability – can you restore the data and the associated files without the software
- Underlying database should have a logical structure and use standards tools, such as SQL for querying the database
- Statistics and reports – both for the public and for the institution

Slide 12 Software requirements

- Flexible Egress – meaning, can you get the files and the data out and port them into a new system
- Flexible ingest – meaning that there are a number of ways that you can get materials into the IR; it can be problematic if the software was set up with one particular ingest or submission model in mind that has been hard coded

Since we were a small group of people who already had many other responsibilities, and since there was the expectation that we would be able to bring up an IR without any additional help or resources – at least in test – we had some additional requirements, which included: **(CLICK)**

- Already implemented in other institutions
- Easy to implement
- Cheap

Slide 13 Metadata support

these are the kinds of issues you'll want to weigh as you plan for and implement a system.

- Our number one consideration regarding metadata was that the system support Dublin Core. All other considerations for us were secondary. So you'll want to consider the type of metadata you want the IR to support, be it MARC, DC, VRA Core, or something else entirely.
- Whether your IR will be a system where you expect to have your users submitting at least some of their own files and contributing their own metadata for those files, or whether you plan to have your staff doing all the submission and assigning of metadata, you may want to consider the ease with which metadata fields can be changed, as well as the interface for correcting or modifying the values given in the fields.
- Is it possible to do any sort of global changes, either across collections or across the entire repository?
- Does the software provide any kind of support for controlled terms, either locally created or using external sources? Does it provide any kind of support for authority control?

Slide 14 Interoperability

When people talk about interoperability in terms of IRs, what do they mean? What is the goal? Since IRs are by their nature usually designed to be open access, a primary goal is to be able to link up with other similar archives. This is done by providing access to at least the metadata, if not to the actual files or content that the metadata describes.

Slide 15 Interoperability: the solution

Open Archives Initiative-Protocol for Metadata Harvesting is a protocol developed by the Open Archives Initiative. It is used to harvest the metadata descriptions of the records in an archive so that services can be built using metadata from many archives. OAI is an XML representation of the Dublin Core metadata set and is traditionally transported directly via http.

You want to establish an IR that is OAI-compliant.

Slide 16 OAI Registries

If your archive is OAI compliant, you can (and should) register it with some of the open archives registries. This greatly enhances the findability of the resources in your IR and also helps your users find other relevant content that might be freely available.

Slide 17 OAI Registries: OAIster

Other registries, such as OAIster, provide the ability to search multiple archives simultaneously.

OAIster is a project of the University of Michigan Digital Library Production Service. Their goal is to create a collection of freely available, previously difficult-to-access, academically-oriented digital resources that are easily searchable by anyone.
I have linked to OAIster from our IR's home page – as I mentioned.

We have also registered with OAIster so that people can find our items here and be led back into our collection of resources.

Slide 18 Version control and revision

One of the questions that our users and potential users are most concerned about is version control.

- They want to make sure that different versions of their work can be correctly identified (metadata can accomplish this)

- They want to make sure that their work is secure and cannot be modified by someone else (checksums are one way of assuring this; as is provenance metadata)

- They want to be able to revise their content themselves.

Many IR packages accommodate the first two of these but don't provide an easy mechanism for the third one.

Slide 19 Public user interface

The public user interface is one of the aspects of an IR that you'll want to consider. You naturally want something that is easy to understand and use and is web-based. Some specific aspects of this that you'll want to look at are the

- Submission template – is this intuitive?; can you modify it easily for different groups or collections?;
- searching – is searching easy? Can you modify the standard search mechanisms that the software provides; can you integrate searches of your IR with other databases your institution maintains
- Does the software provide any tools that facilitate the capturing of metadata for citations or the reuse of content (when permission for its reuse has been granted)

Slide 20 Digital preservation

I'm going to try to give a whirlwind glimpse of some of the issues underlying digital preservation so that you can appreciate something of the complexity. I've attended 3 weeklong digital preservation workshops around the world and I'm going to condense that down into 5 minutes. **(Continued ...)**

Preserving digital content entails far more than making backup copies and storing them in disparate locations. Digital preservation is a series of managed activities necessary for ensuring both the long-term maintenance of the files and continued accessibility of their contents.

Designed to extend the usable life of machine-readable files and protect them from media failure, physical loss, and hardware and software obsolescence, these activities include:

- ensuring the long-term maintenance of a bitstream (the zeros and ones):
    - backing up files and keeping a copy at an offsite location
    - running checks to track the deterioration of storage media, files or bitstreams

Slide 21 Digital preservation

- Providing continued accessibility of the contents:
    - *viability* - making sure that information is intact and readable from the storage media
    - *renderability* - making sure that information is viewable by humans and able to be processed by computers
    - *understandability* – making sure that information is able to be interpreted by humans

Slide 22 Digital preservation strategies

There are many strategies being discussed and employed for preserving digital content. This list is taken from ***some*** of the options outlined on the Cornell digital preservation tutorial. In order to decide on your strategy, you need to be clear about what you're trying to accomplish and then plan according to your resources.

I won't go into what any of these mean unless you have questions and instead I refer you to the Cornell tutorial which is on your references and sources handout

Slide 23 Digital preservation components

There are some components that are important for digital preservation that you should make sure that your IR can accommodate, at a minimum

Slide 24 Metadata registry

This is a screen shot from the metadata registry in the UO's IR that uses DSpace software. What metadata fields do you support in your IR and what are your input standards and mapping for those fields. This might also be handled through an external data dictionary, although it's preferable that your IR software provide this.

Slide 25 Format registry

A format registry will help you track the file formats that your software system can support technically – but you will still have to make policy decisions about which formats you will preserve and how you will try to preserve them.

Slide 26 Checksum verification

Checksums are one option – this is an algorithm which provides a unique identifier for a digital bitstream.

If this is automatically generated at the point of ingest or submission of files, then you have a basis for comparison if you re-apply the algorithm later on to let you know that the file has either deteriorated or been tampered with

Slide 27 Backup procedures

You will need to establish regular backup procedures and then document them.

Slide 28 Persistent identifiers

You'll want something that provides a uniform resource identifier (such as a URL or a handle) – something that can be cited and linked to, at various levels (such as community, collection, record, bitstream)

DSpace uses the Corporation for National Research Initiatives handle system, which is a comprehensive system for assigning, managing, and resolving persistent identifiers, known as "handles," for digital objects and other resources on the Internet. Handles can be used as Uniform Resource Names (URNs).

There are other types of persistent identifiers but this is an essential component of a digital preservation strategy.

Slide 29 Attributes of a TDR

A trusted digital repository is one whose mission is to provide reliable, long-term access to managed digital resources for a designated community, now and in the future. Its attributes are outlined here but I will leave you to explore this concept further by checking into some of the sources I've listed for you.

Slide 30 OAIS Reference Model

The OAIS (Open Archival Information System) Reference Model was developed by NASA's Consultative Committee for Space Data Systems to "create a consensus on what is required for an archive to provide permanent, or indefinite long-term, preservation of digital information."

It is becoming the accepted conceptual framework for an archival digital preservation system and has been used by OCLC and RLG to develop the framework for Trusted Digital Repositories (TDR). The model attempts to identify the responsibilities and components of an archival system

Slide 31 Test site

It’s highly recommended that you maintain a test space for your IR that is different from your production site. A place where you can test out new software versions as well as try out new approaches. Every time there is a new release of the DSpace software or a significant bug fix, we test it here before we implement it on the production server.

This is also where I set up new communities and collections and give the community liaisons a chance to react to a proposed structure before going live with it. This saves you a lot of time and wasted energy in the very short run, as well as the long term.