More thoughts on thesis embargoes

I wrote last time about how we might do thesis embargoes with ICE as part of the TheOREM-ICE project we're doing with Jim Downing and team at Cambridge. That post was mostly about why we wouldn't want to add complex access control at a very granular level to ICE.

I'm actually in Cambridge now and I've talked the issue over with Jim Downing and Nick Day. We think we've come up with a workable, implementable prototype for thesis embargo which I'll describe here. But first some background about the requirements.

We've been whiteboarding thesis workflow with three broad stages;

1. Dafting, where a small group have access to the emerging document plus its data. The group could be as small as one candidate and one supervisor but others may be allowed in. We don't need fine grained access control for this bit and embargoes are not relevant.
2. Examination, where the thesis goes off to examiners. Again you don't want to embargo anything or what would they be examining? There will be some indication of which bits are *going to be* embargoed though so examiners can take care not to talk about those bits to others. Access control is crucial, but so is management of the examiners and managing their feedback. It's not clear that we would want to add these features to ICE, maybe to OJS?
3. Making public, where we do need to worry about embargoes. At this stage the content should be out of the drafting system; this is what the ARROW people called crossing the curation boundary.

While we were talking about this it came to light that one of the theses that we're looking at for the TheOREM-ICE project has a rather unusual embargo requirement. The thesis is OK to go on the web apart from the acknowledgements section which is apparently considered by the author to be a private matter. The solution we came up with will deal with that.

Here it is:

In ICE each chapter/part of the document will be a separate document, in Word or OpenOffice.org writer, as per the way ICE does courseware. This is much safer for book-length content than trying to use Word on a single file, anyway.

The candidate will enter embargo information into each chapter using the techniques outlined in our joint paper that Jim Downing recently delivered at the IDCC conference in Edinburgh. That is, at the top of each file will be a place to put some embargo information. We have to work out how this will look but it will be something like:

Embargo period (in months) • Leave blank or put zero for no embargo • Put -1 for indefinite embargo	6
Embargo manager (optional)	http://ptsefton.com/pt

This is human readable, so it will give cues to supervisors and examiners about how the data are to be treated, but it will also be machine readable so that downstream systems can be used to enforce the embargos.

The downstream system we have in mind to demonstrate this is The Fascinator, which was designed for just this sort of thing.

Once the thesis is awarded, someone in authority will be able to click the 'Make Public' button, which

will push the content to an instance of The Fascinator using an OAI-ORE resource map to describe the contents of the thesis in excruciating detail including the metadata about embargoes. The Fascinator will index the embargo dates and enforce access controls using its normal approach of limit queries. That is, the guest account will always have a query like this added:

```
issue_date<=$todays_date and total_embargo=false
```

This says make sure that we only show things issued on or before today which are not under unconditional embargo. At the moment it seems like the simplest solution for embargo dates will be to use the Dublin Core issue date element. If an item has an issue date in the future then The Fascinator will refuse to serve it to guests.

The openId field in the metadata example above is a place to capture an identity that is controlled by the candidate so they can come back and manage embargoes later, after their local institutional login may have been turned off. We don't know how to store that metadata just yet.

We may not be able to implement all of this immediately because we're coming up to the silly season in Australia, but I'm sure that we will be able to demo parts of this scenario fairly quickly.