ABSTRACT

Many institutions have addressed administrative policy application, asset management by group/collection, administrative roles and related issues in the process of developing the Hydra functionality necessary to meet their institutional needs. For various logistical and technical reasons this development and the resulting code have often been localized for the specific institution. Although these solutions have met the needs of the time, we (I, Mark) believe that some common principles and best practices have emerged.

ASSERTIONS

• Models for arbitrarily flexible administrative policy exist within a variety of computer file systems which could be used as models for a policy system for Hydra
• Based on anecdotal experience the more variety and exceptionality supported in a policy system, the more complex it becomes to manage (are there evidenced based studies on this?)
• Policy systems which allow nested, grouped, or cascading policies with inheritance can lead to exceedingly complex policy rules which generally require complex analytical tools to understand and cumbersome administrative tools to manage.
• Inheritance chains are generally opaque to content owners who may not understand why/how to grant/deny access to their content within a repository environment.
• Corollary: changes to policy at the group level may have unintended outcomes at the item level
• Library and Archive collections frequently have large number of exceptions to standard patterns for access and administration - e.g. unique embargoes, personally sensitive information mixed in with open content within a larger archive accession, restriction differences on master content vs. derivatives.

COUNTER ASSERTIONS

• CSS represents a similar nested, grouped, and cascaded policy system - we all get along fine with that (however the difference may be that errors in CSS policy inheritance are usually obviously and visibly identifiable, and errors in display formatting may be more acceptable than errors in content access or modification restrictions)
• A repository hosting any sizable amount of content requires some form administrative partitioning and delegation to support effective management by multiple parties.

RECOMMENDATIONS

• Access and administrative policy should only be applied at the atomic object level
• Policies can (& should usually) have a one-to-many relationship with atomic objects
• Objects should permit only a single policy to be applied at any given time
• Policy should be applied to objects in collections or other administrative groupings should be applied by batch processes which update the local policy stored on each atomic object
• Tools should be provided to analyzed
  • How long policy application might take for a given batch
  • What exceptions would occur for a given batch