Push vs. pull

- Library-centric ("pull")
  - library actively obtains, processes data, metadata
  - providers need only give access
  - MIL approach to date
  - result: *N* archived collections
- Library-hosted ("push")
  - library provides, hosts archival services
  - providers use services, integrate use into operations
  - result: archival infrastructure

Pull

- Pro
  - requires little/no cooperation from providers
  - proven
    - e.g., Internet Archive
  - greater centralized control, consistency
  - less risky

- Con
  - poor scalability
  - incomplete/inconsistent view of provider content

Archival interface

- Archives themselves are opaque
  - simple
  - depository philosophy
- Interface characteristics
  - defines object model
  - handles PIDs/GUIDs, checksums/signatures
  - semi-formalizes handling of IP concerns
- Questions
  - what services (delete, replace, etc.)?
  - inter-object relationships (supersedes, etc.)?

Object model

- Take advantage of geospatial focus
  - complex, but known object structures
- Data
  - files
  - standardized structuring (e.g., METS)
  - file relationships
- Metadata
  - multiple, heterogeneous records
  - mappings to ADL views
- Links to semantic definitions

Metasupport

- Registries
  - file formats, format definitions
  - metadata formats, metadata semantics

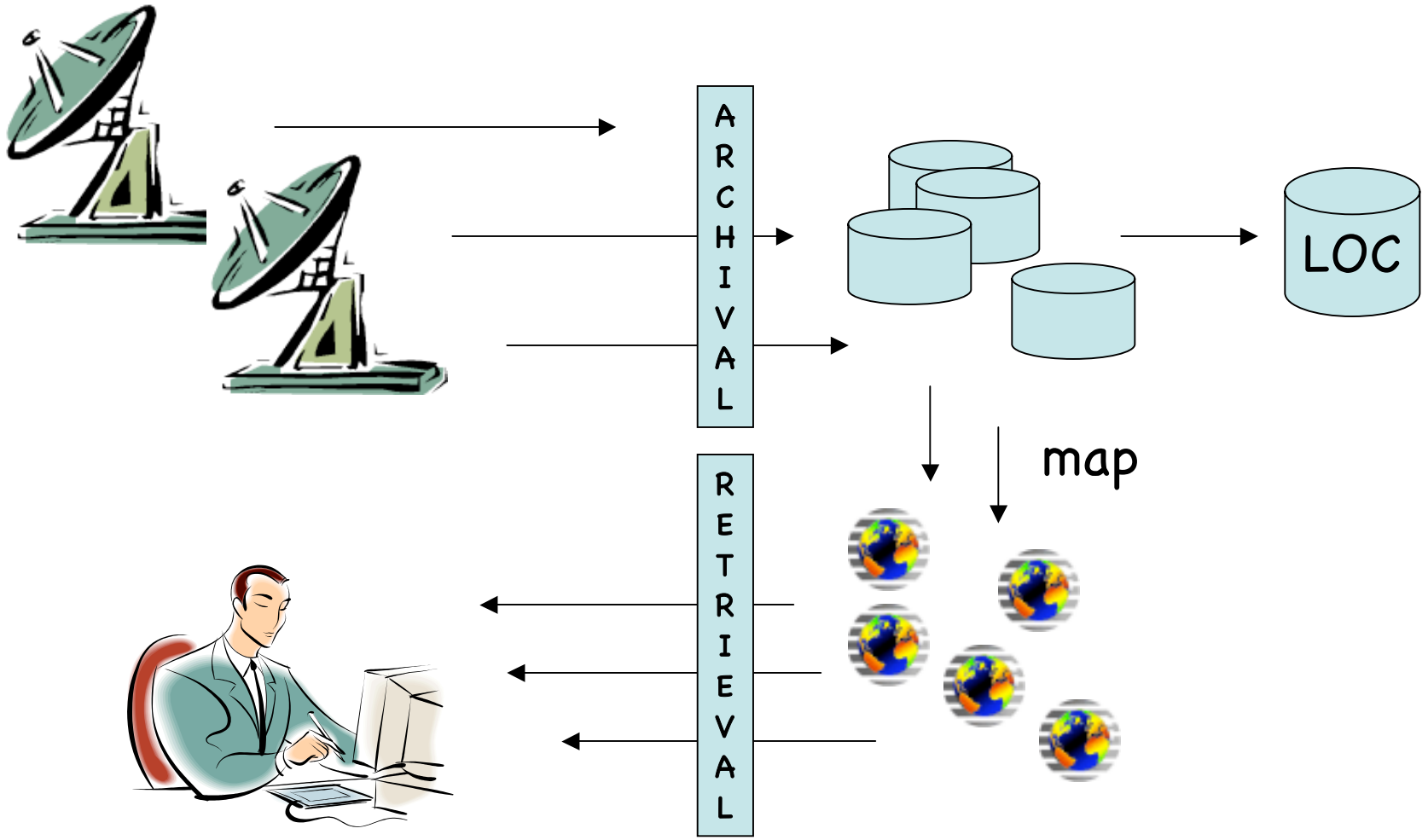
- (aside: Fedora?)

Archive implementation

- Lots of flexibility

- Time to explore interesting ideas
  - federations of archives
  - mirrors
  - automatic transfers to LOC

Big picture



Development risks

- Low
  - archive implementation, ADL mapping
    - hidden, known
- Moderate
  - archival interface, metasupport
    - providers, software rely on it
- High
  - getting users
    - need 1+ big, external providers