Validation of RPKI objects using a local cache
Problems with current

• Very tight coupling to rsync
  - Need to process objects not on manifest
  - Vulnerable to updates happening during fetch

• Prefix validate wants to know all ROAs

• Implementations use URI as identifiers for objects
  - Multiple publication points complicated
  - Same for alternative fetch mechanisms
Decoupling object retrieval

• Use SIA, AIA and CRLDP only for object discovery

• Allows for other retrieval mechanisms
  - rsync
  - bittorrent
  - http with / without deltas
  - multiple publication points
  - other..
Validation using ‘just objects’

find by:  
*Key Identifier*  

find by:  
*hash*

- **TA Cert**
  - SKI

- **MFT EE**
  - AKI

- **CRL**
  - AKI

- **CA1 Cert**
  - SKI

- **CA2 Cert**
  - SKI

- **MFT EE**
  - AKI

TAL

*latest?  
signature ok?  
all objects?*

*there can be only one...*
Differences from current RFCs

• Strict interpretation of current repository standards
  – Some clarification for CAs might be useful:
    MUST 1 mft, 1 crl, all objects that need to be known

• Manifests authoritative source for walking the tree
  – Ignores objects that the CA does not put on mft
  – May be strict if objects are missing, e.g. go with last known
good state if available

• SIA, AIA and CRLDP only for discovery