

# **Constructing a Records Archiving System Using Off- the-shelf Tools - A Lightweight Approach**

**IWAW  
2007**

**Jan Askhoj, Mitsuharu Nagamori,  
Shigeo Sugimoto**

**Tsukuba University, Japan**

# Outline

- Problems with corporate archive systems
- Our Goal
- Proposed Model
- Use Case
- Technical Details & Implementation
- Current work & Concluding remarks

# Introduction

- For corporations today, bad records Management is not an option!
- There are a lot of standards and laws to adhere to...



# Problem

- Records Management Systems (RMS) are used to ensure compliance

“- Gartner estimates the worldwide records management market to be approx. \$350 mill. in software license and maintenance revenue in 2006, up more than 30% from 2005”

- Systems used to archive content are seldom good for creation... and vice versa

# Furthermore...

- Many kinds of Content Management Systems in use:
  - Web Content Management
  - Digital Asset Management
  - Document Management
  - etc.
- All these may contain content that needs to be archived !

# Our Goal

- Model for archiving corporate records:
  - Lightweight
  - Automated
  - Flexible
  - Secure



Content +  
Metadata

Archive

# Ideal Situation

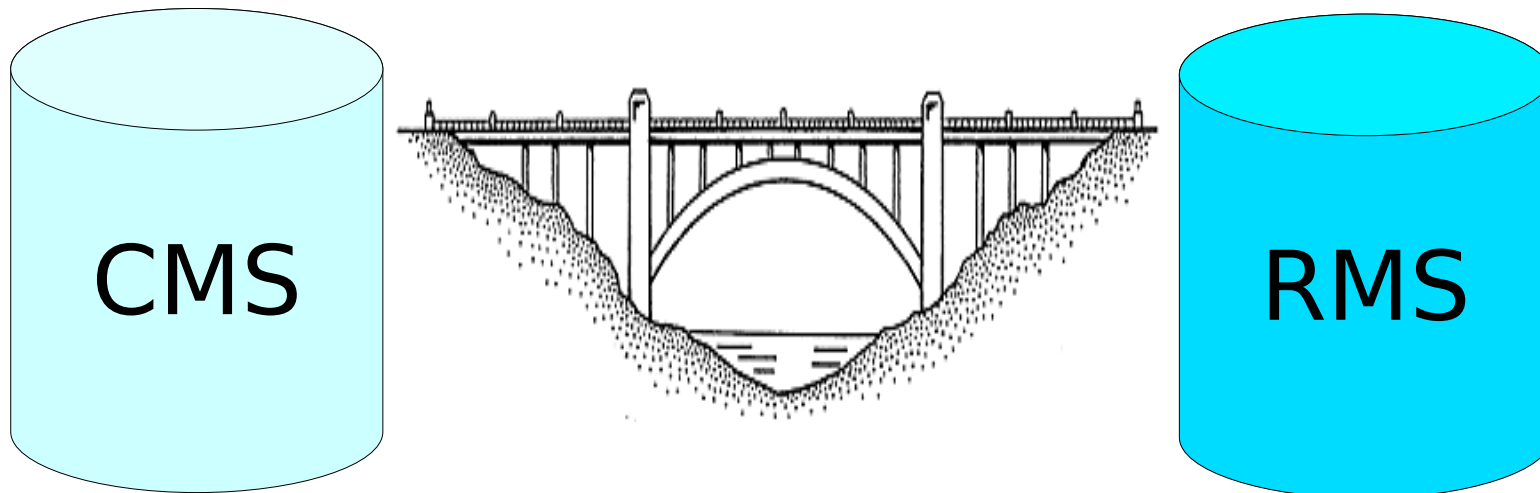
- A system to manage the entire document lifecycle



- To keep cost and complexity low:  
move data from CMS to MS
- This cannot be done in the box with  
different systems!

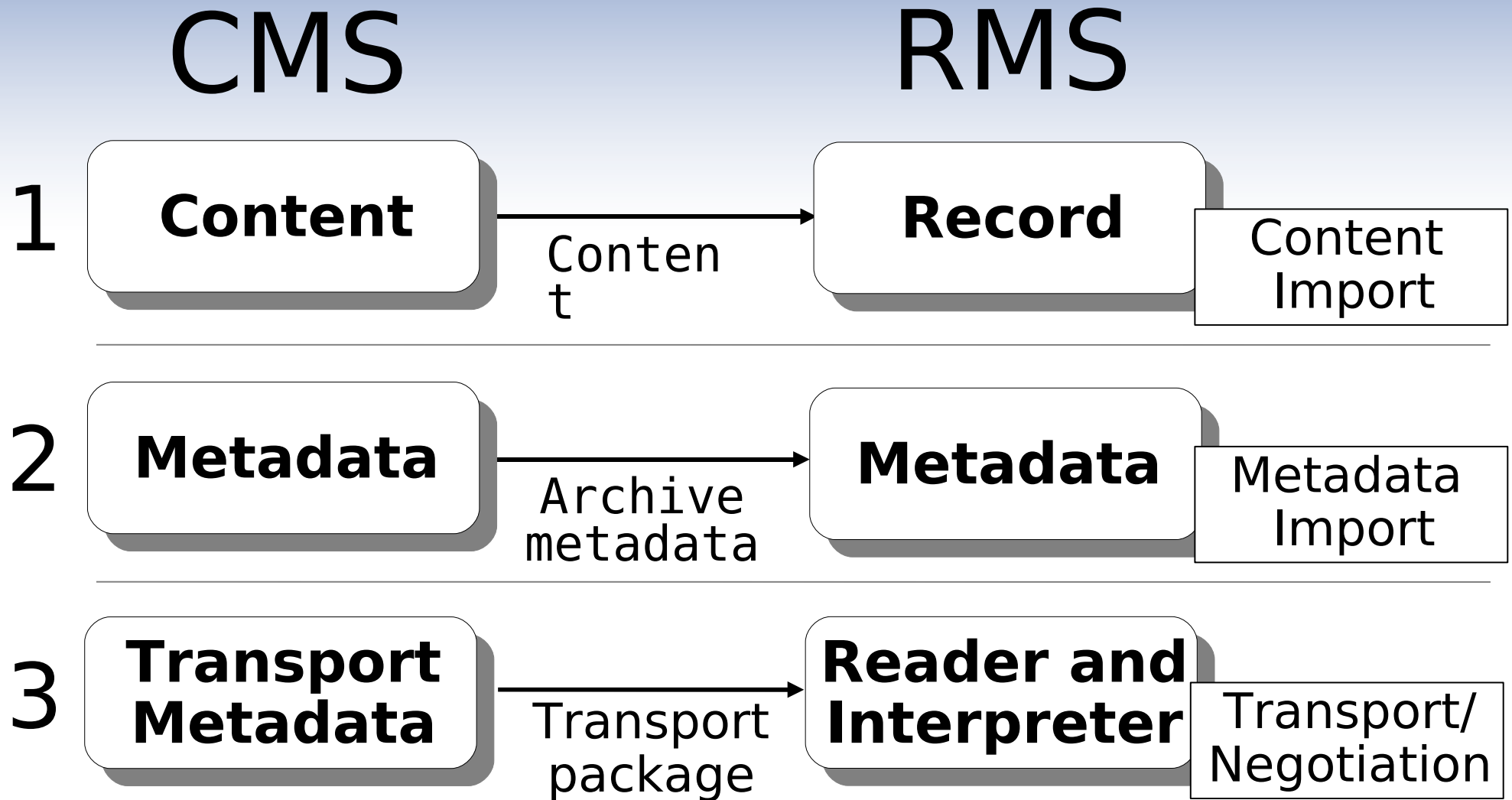
# Bridging the GAP

- Moving Records:
  - By hand ?
  - By custom export-import scripts ?
- A model is needed for CMS→RMS archiving (that works with exiting systems)





# Three Layered Model



# Information Flow: Export

## CMS

**Content**

- User submits content
- Retention by content type

**Metadata**

- Metadata extracted

**Transport  
Metadata**

- Metadata + content packaged
- Transport Package made available

# Information Flow: Import

- Content is approved and imported
- Metadata is approved and imported
- Package is read
- Metadata is converted
- Trackback is returned

**RMS**

**Record**

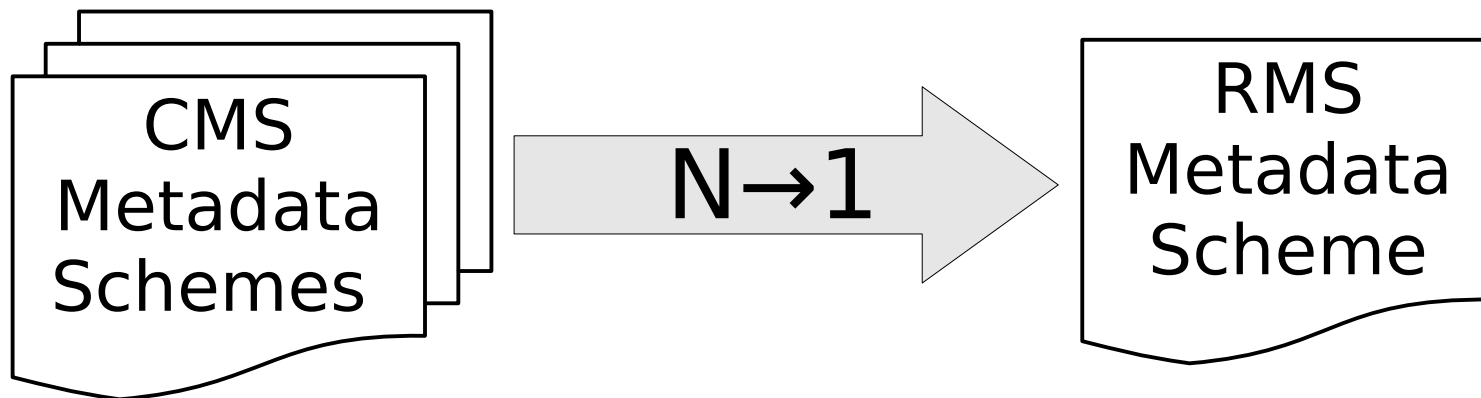
**Metadata**

**Reader and  
Interpreter**



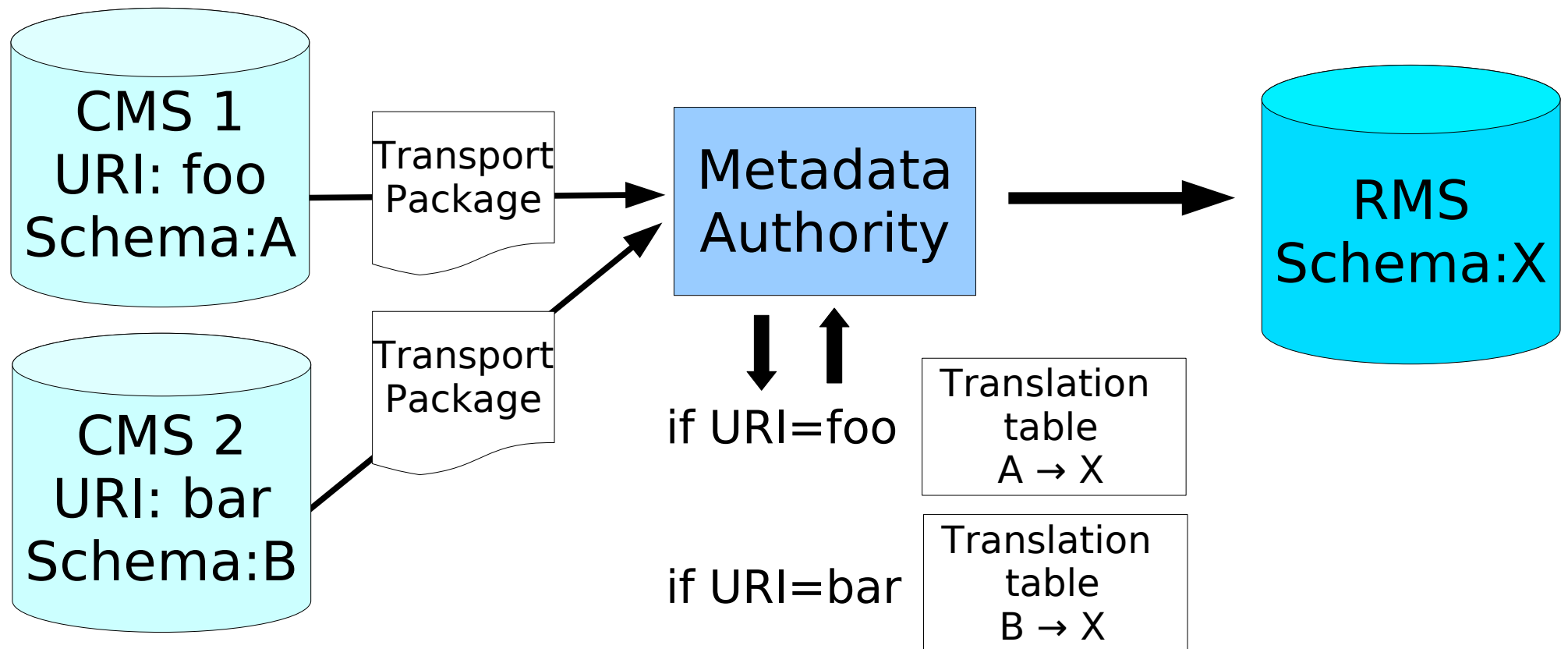
# CMS/RMS Metadata Schemes

- CMS metadata cannot be used by the RMS as-is
  - - CMS metadata schemes:  
Can vary according to application or content
  - - RMS metadata scheme:  
Designed for Records Management

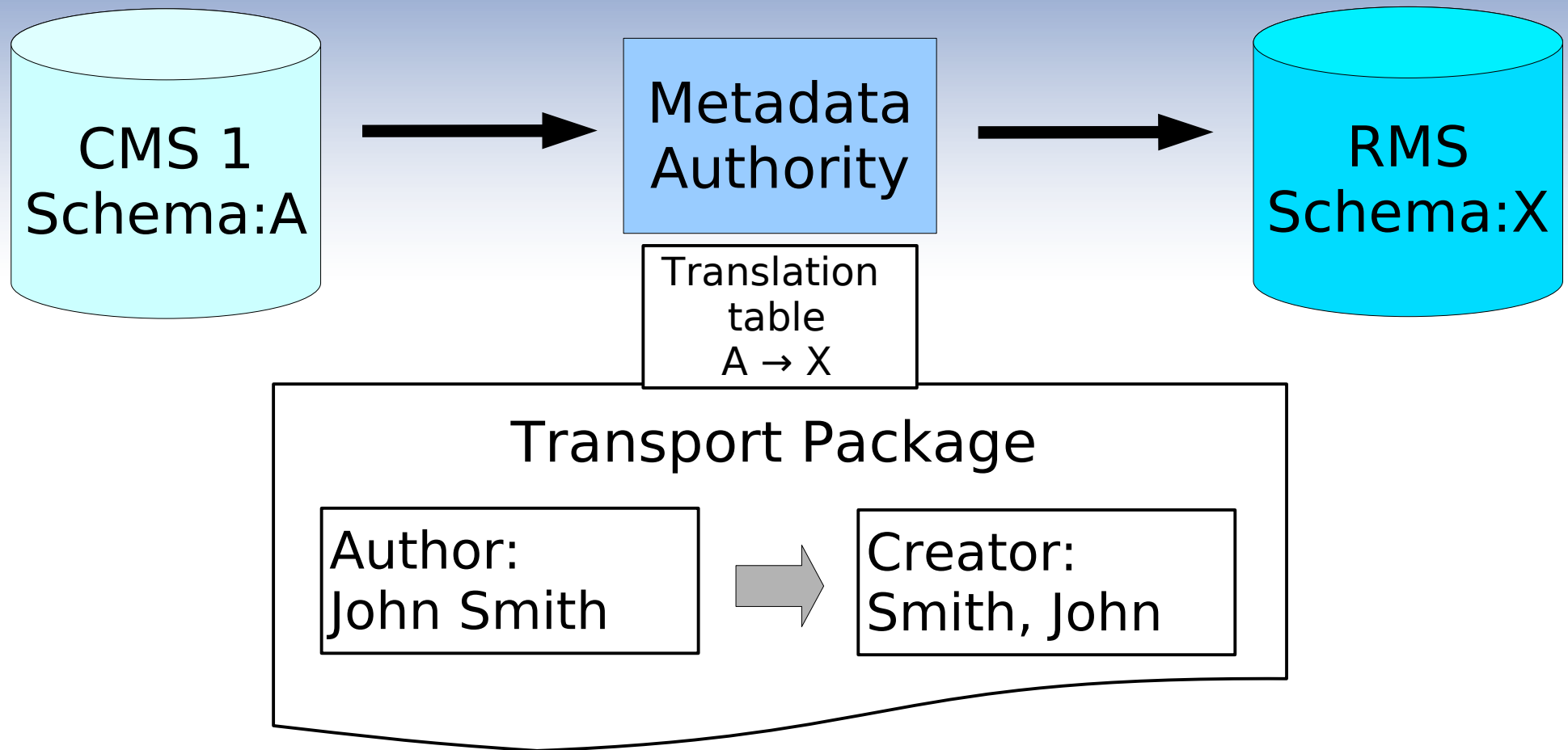


# Metadata Authority

- Offers metadata conversion
- Requires that CMS register URI and Scheme in advance

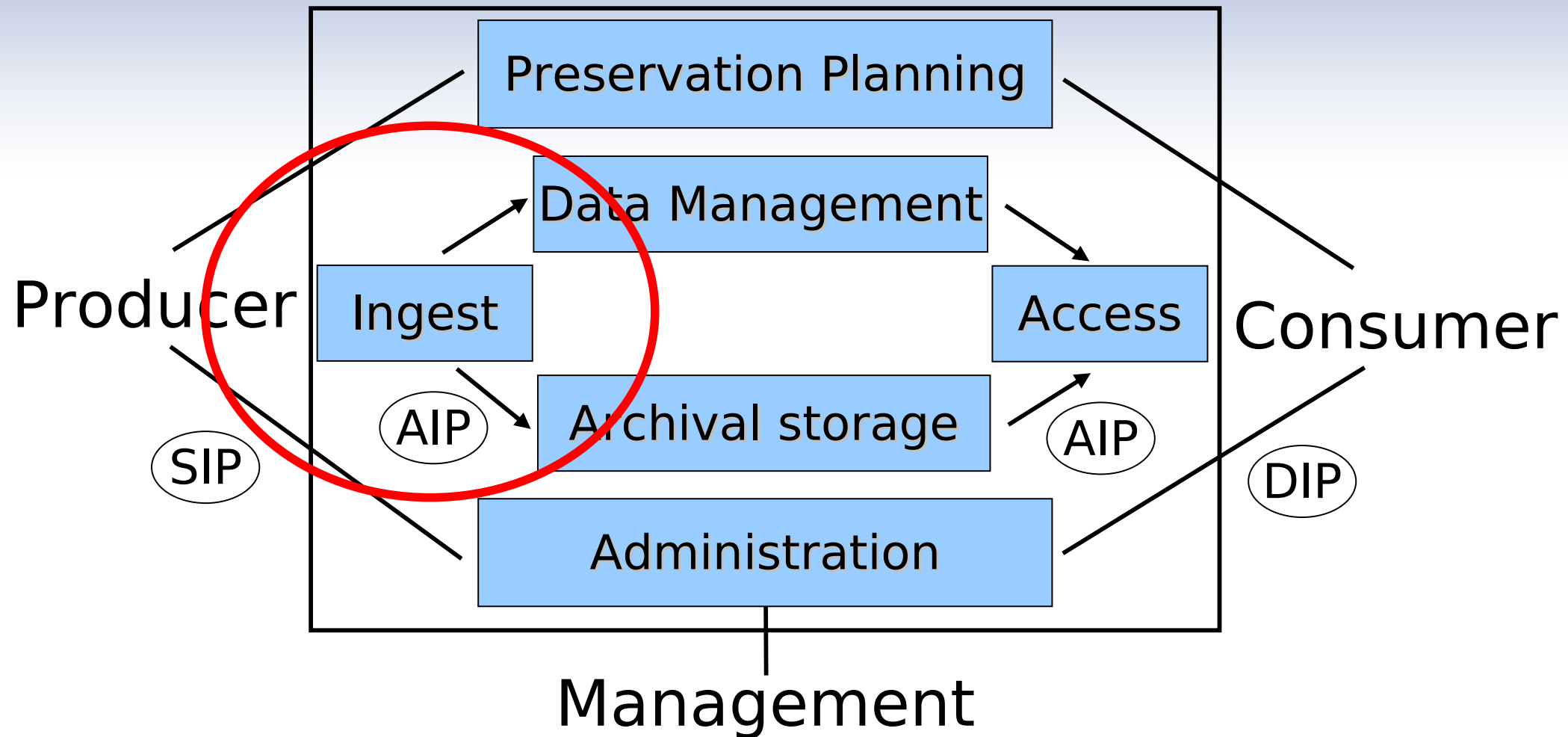


# Example : Metadata Conversion



- Only works with compatible metadata

# Related Models - OAIS



# Implementing ATLAS

- Automated Transfer Lightweight Archive Solution
- Existing System Requirements:
  - CMS: Expandable, publish necessary content and metadata
  - RMS: Import functionality, supporting XML Metadata

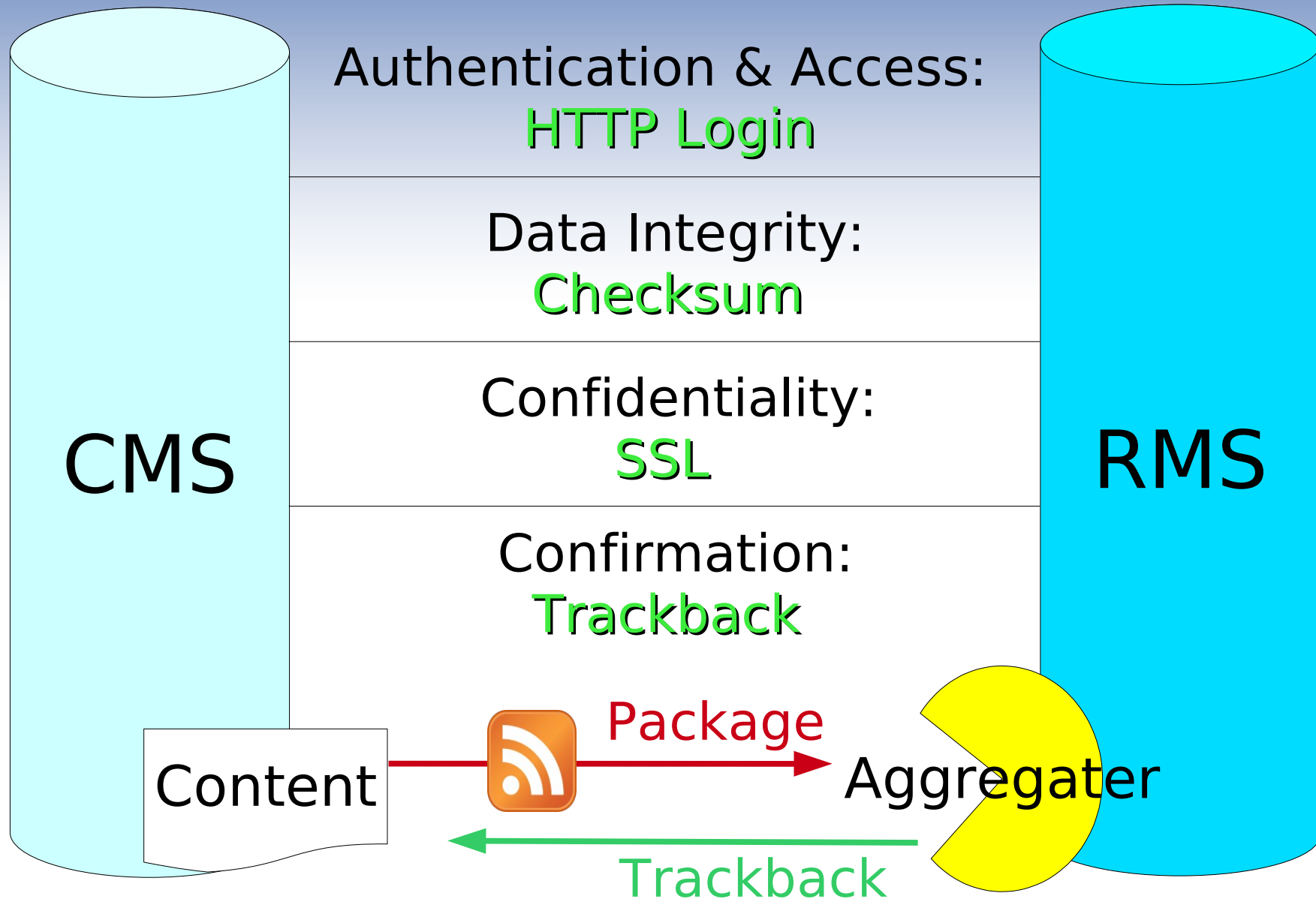


# Protocol

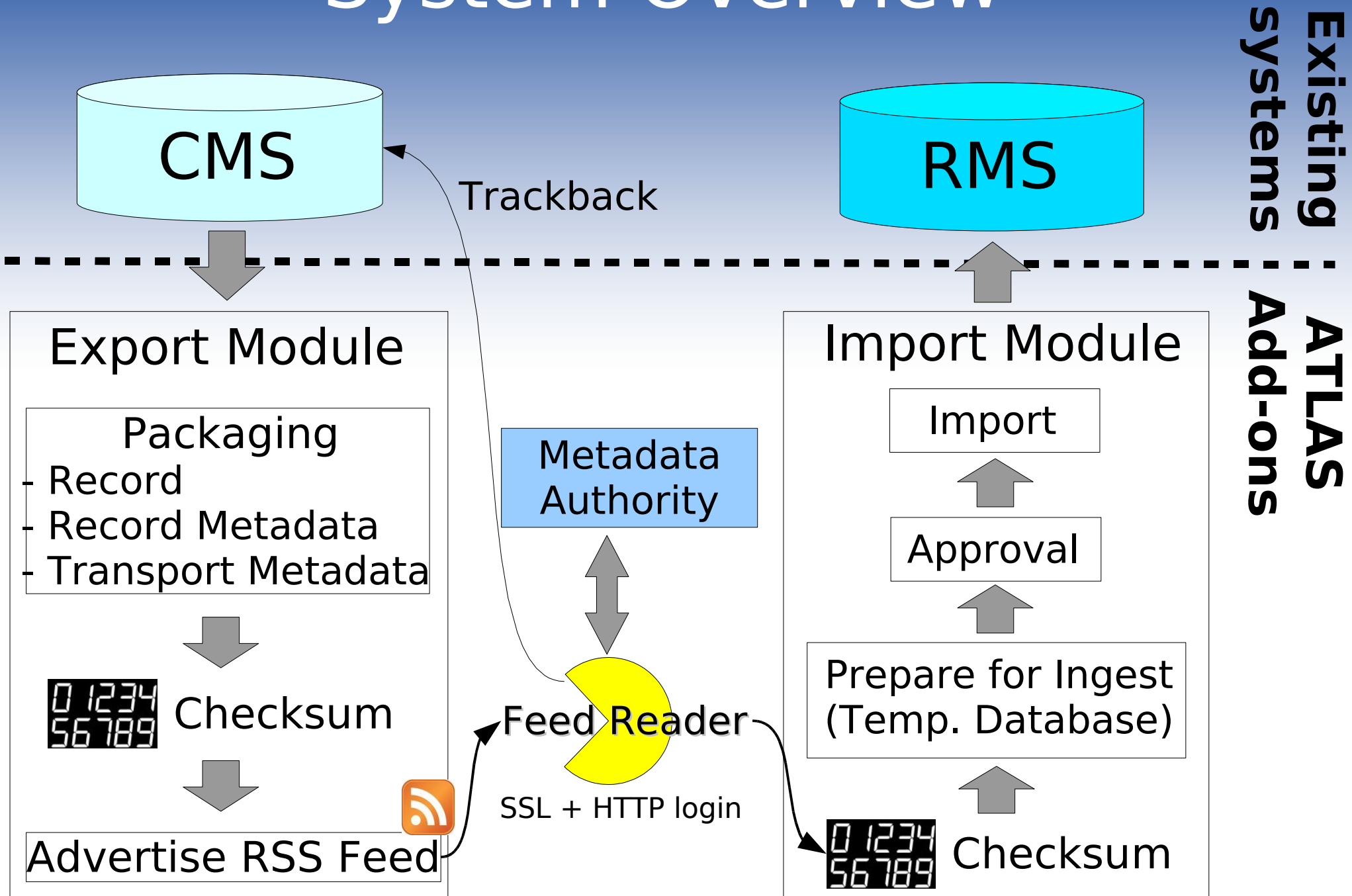
- RSS 2.0 (Really Simple Syndication)
  - XML based
  - Lightweight
  - Expandable
  - Already supported
- Need to Expand: Metadata, Metadata Authority URI, Security



# Security & Trackback



# System Overview



# Issues & Current Status

- Issues
  - CMS must have good metadata
  - No provision for long term-storage
  - Requires good archiving environment
- Current Status
  - Building ATLAS using...
    - CMS: Drupal
    - RMS: Dspace
    - Ruby on Rails

# Concluding Remarks

- Model suited to problem
  - Simple to implement
  - Reduced cost
  - Open for expansion, modification
  - Guarantees safe transfer

Thank you !

Merci beaucoup !

Mange Tak !

ありがとうございました！