



# **Tools: How to Integrate the Components of Digital Preservation**

Dr. Ross King  
Austrian Institute of Technology



# Outline

---

- Definitions
- OAIS Model and Workflows
- Workflow Templates
- Case Study
  - Necessary Steps
  - Workflow Description
  - Putting it all together
- Conclusions



# Definitions

---

- **Workflow**

A Planets preservation workflow is a sequence of Planets Services (which are Web Services that implement one of the specified preservation interfaces like *Identify*, *Modify* or *Migrate*), in which the output parameters of a given service are validly mapped to the input parameters of the following service.

- **Workflow Template**

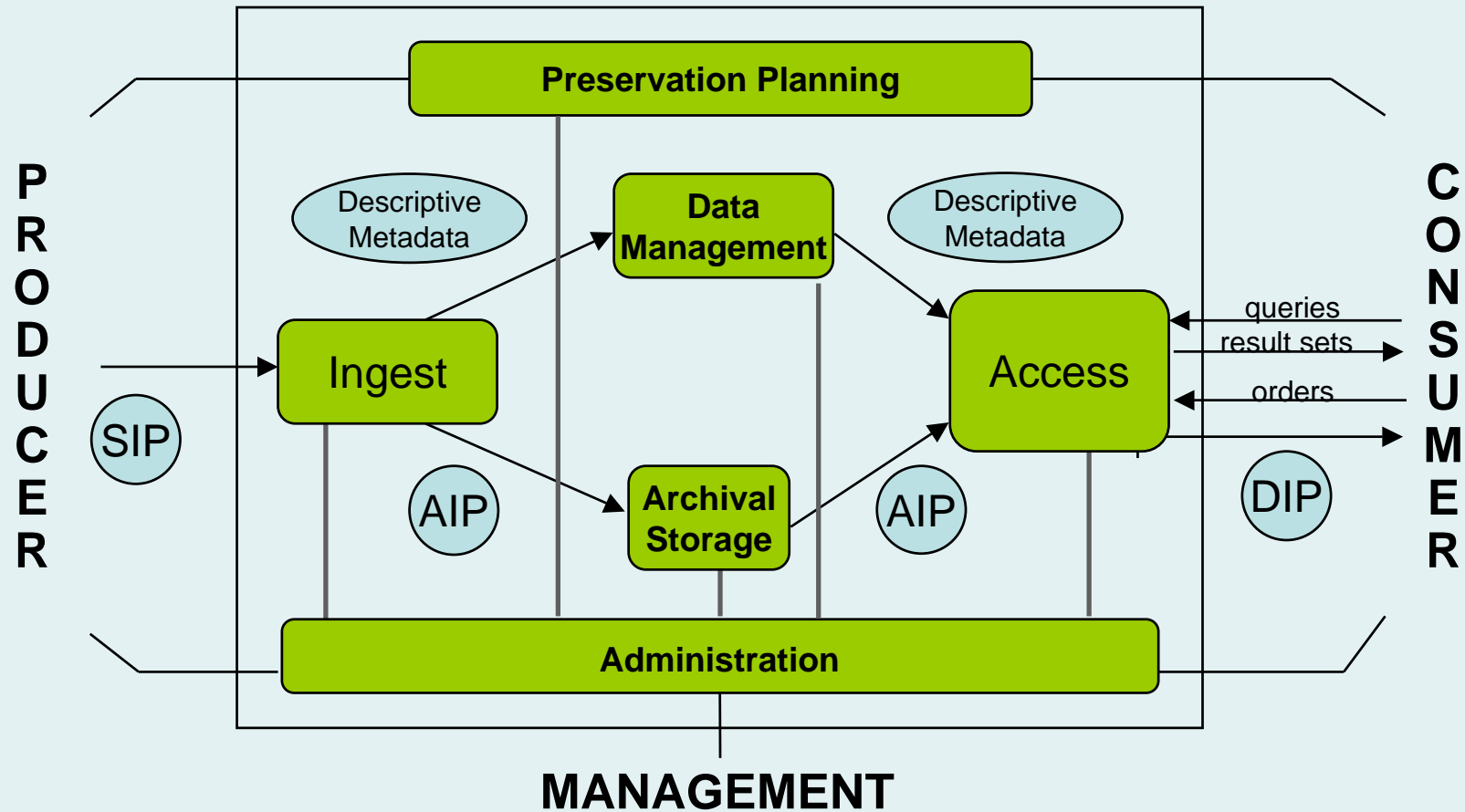
A workflow template is a workflow in which the nodes of the preservation sequence are service placeholders rather than service endpoints. A service placeholder defines only the interface, and the actual functionality behind the interface is irrelevant.

- **Workflow Description**

A workflow description is an XML-serialization of a Planets workflow, which identifies a workflow template, the service endpoints associated with all template placeholders, and the parameters associated with each service.



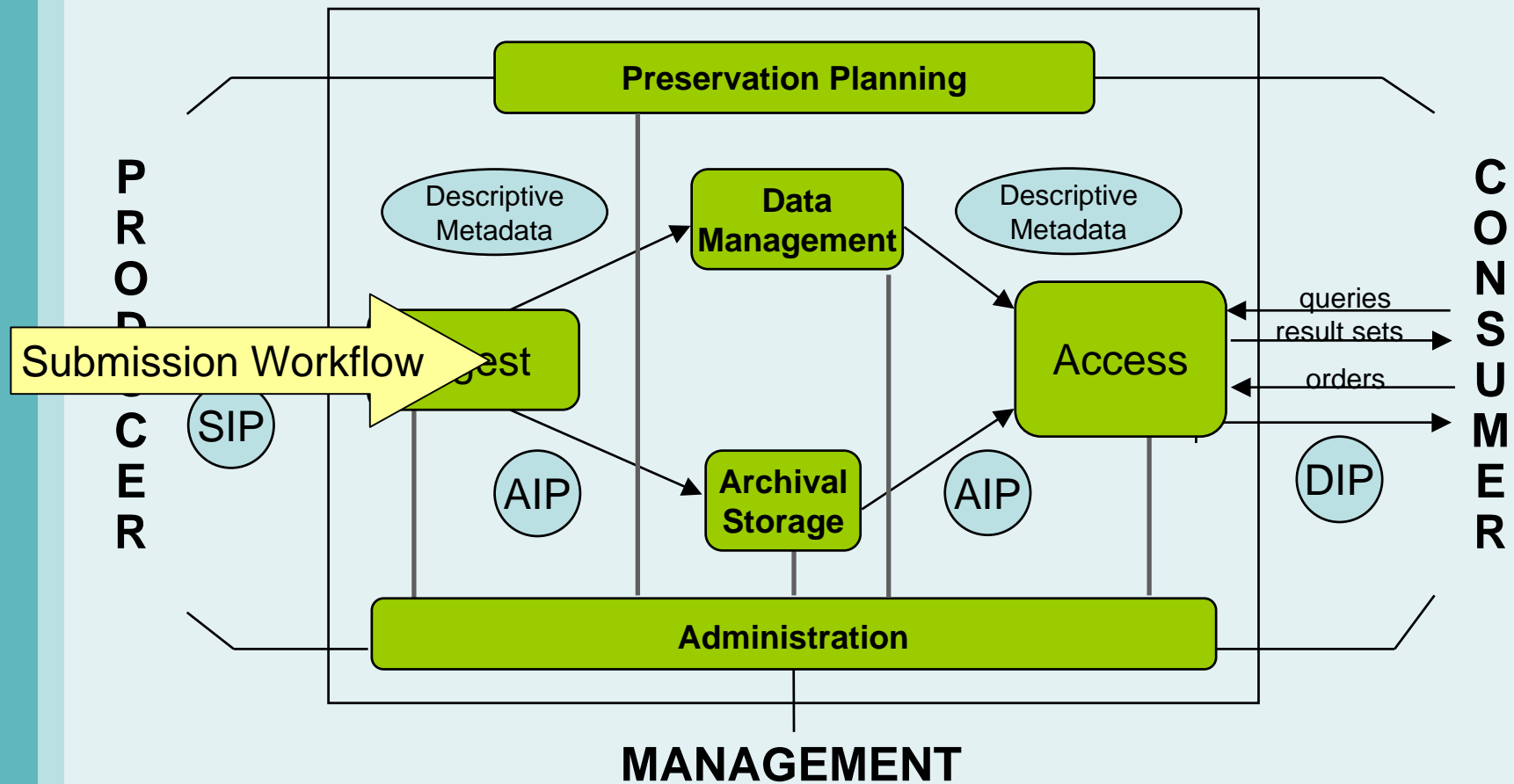
# OAIS Repository Model



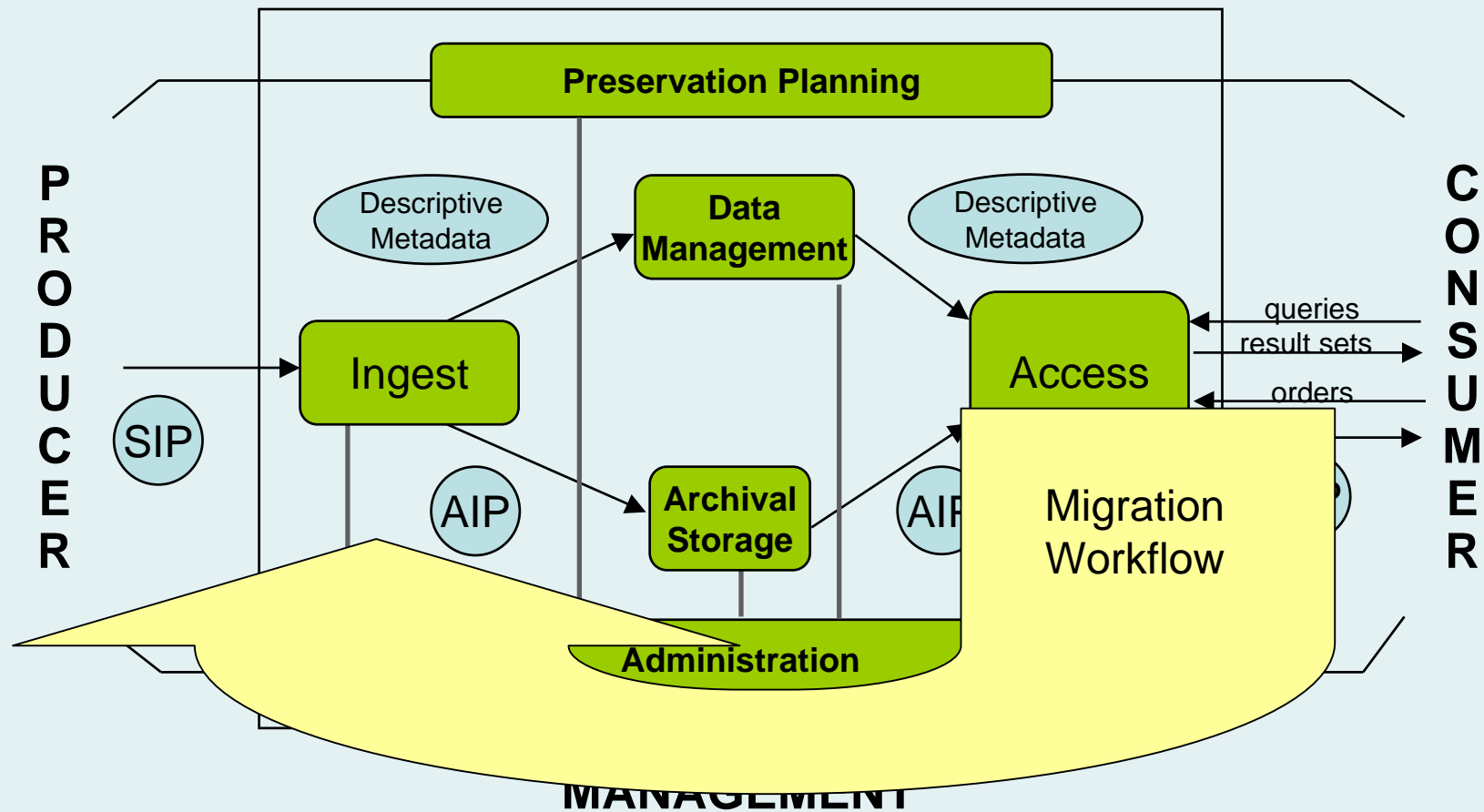
SIP = Submission Information Package  
AIP = Archival Information Package  
DIP = Dissemination Information Package



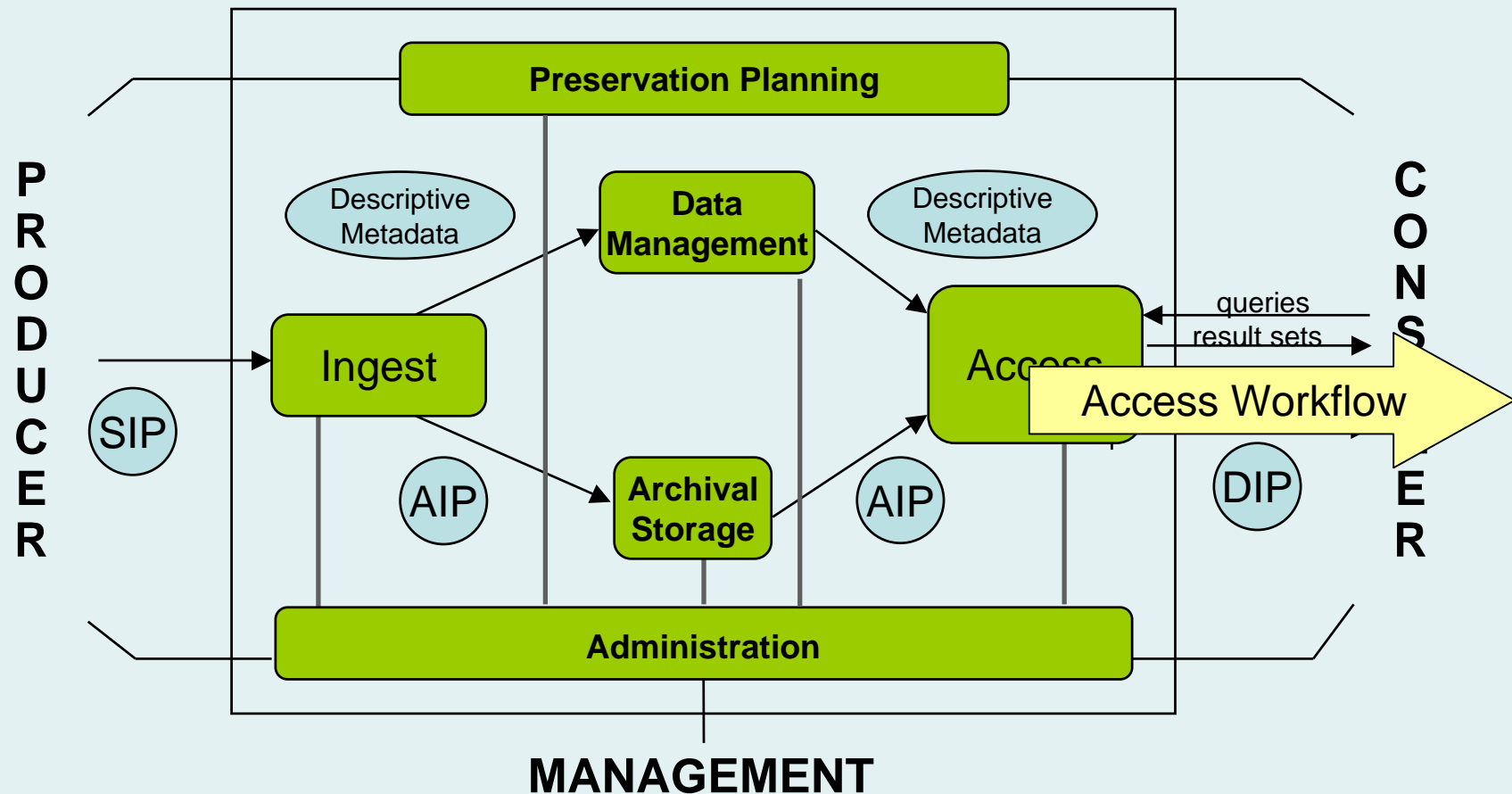
# OAIS Repository Model



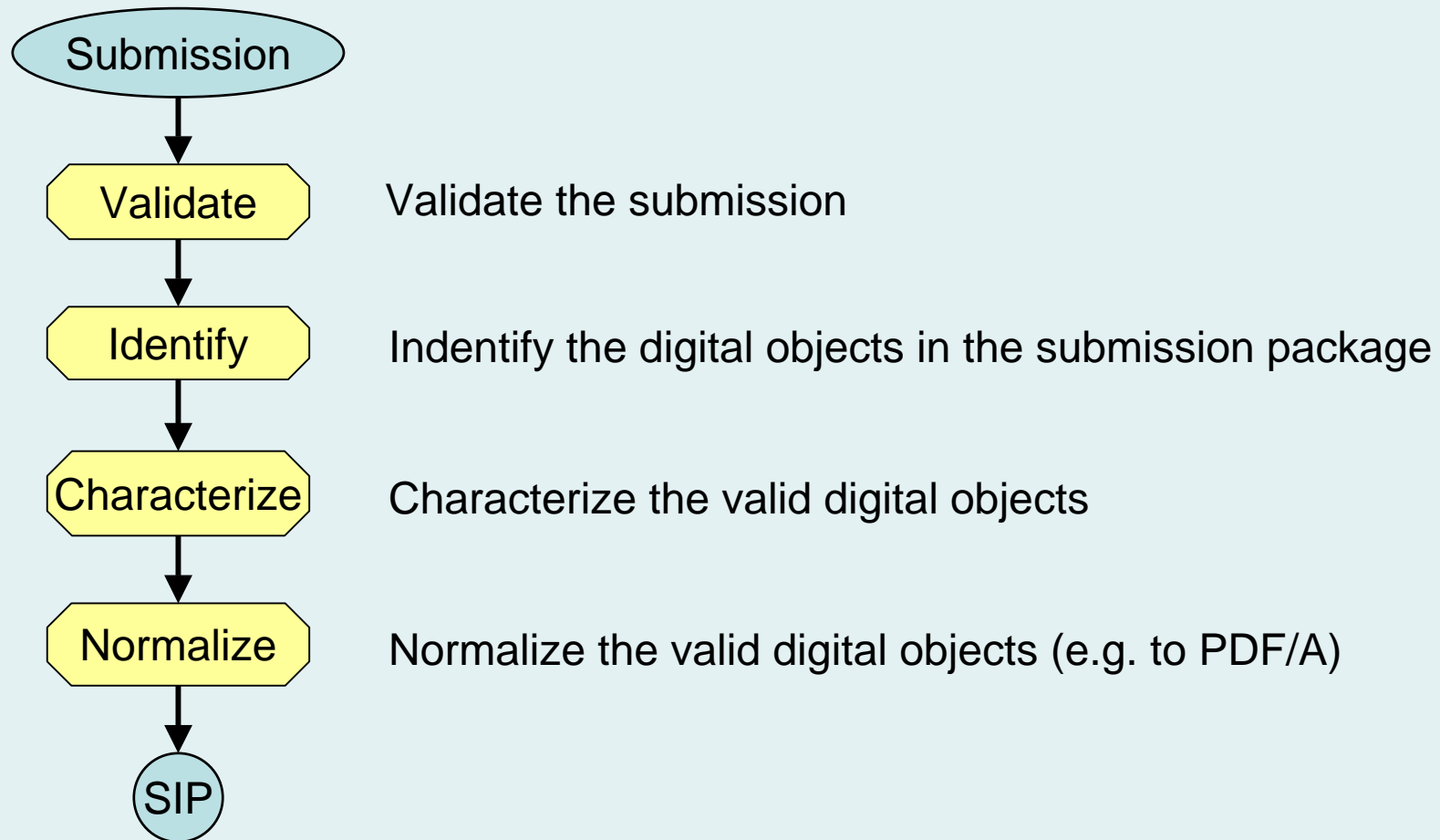
# OAIS Repository Model



# OAIS Repository Model

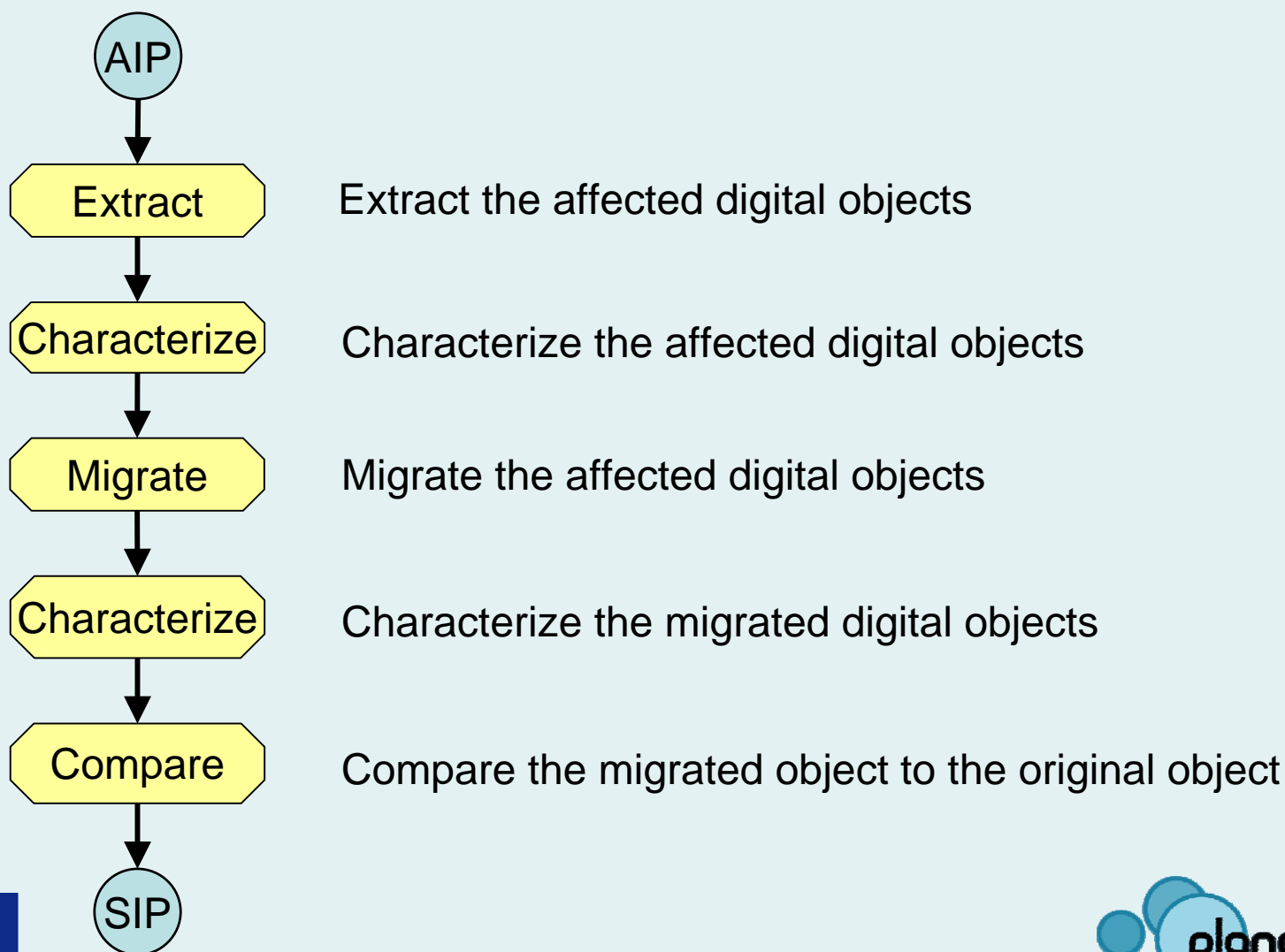


# Submission Workflow Template



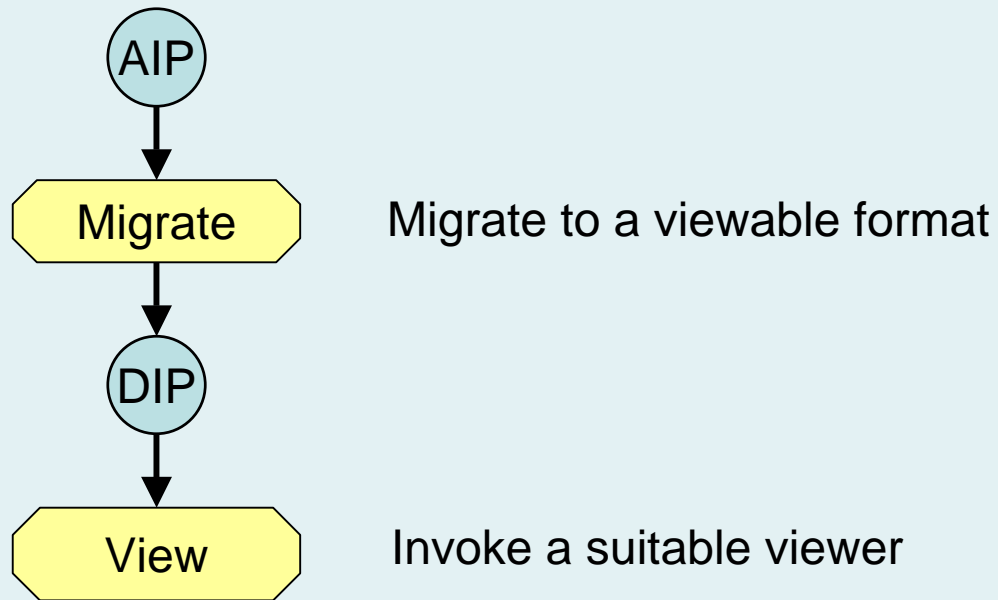


# Migration Workflow Template

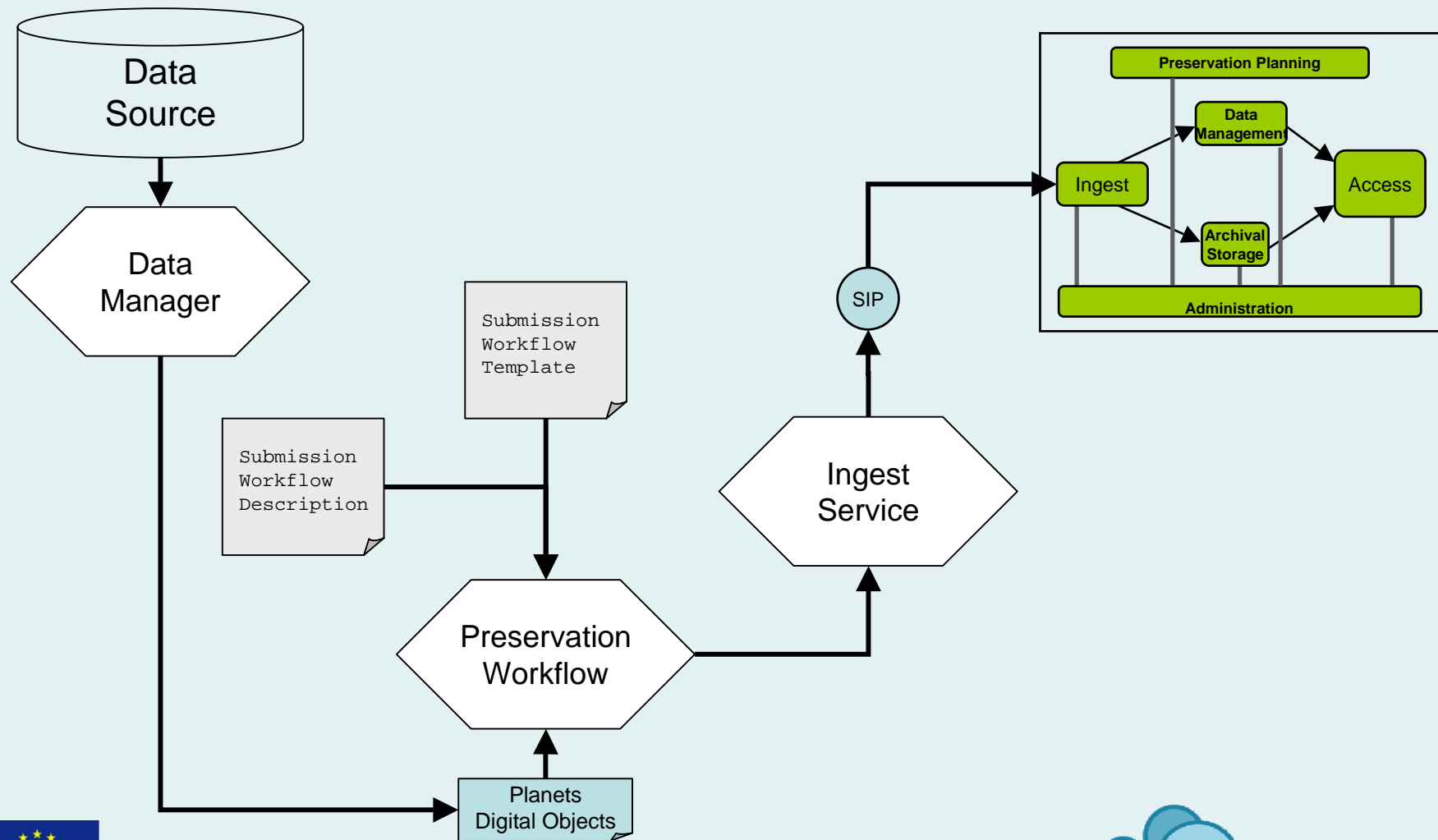


# Access Workflow Template

---



# Repository Integration: Submission



# Case Study: Submission

- The British Library has a large collection (80 TB) of TIFF images (scans of newspapers) that should be placed in archival storage
- A submission consists of a TIFF image and a separate XML descriptor
  - This step requires a custom data manager for this type of content
- The image should first be cropped and rotated according to the descriptor
  - This step requires a slight modification of our generic submission workflow template
- Then the image should be normalized to the JPEG2000 format.



# Repository Integration: Necessary Steps

---

1. Implement a Data Manager
  - must harvest data from your repository or other data sources
2. Choose a Workflow Template
  - or modify an existing template if necessary
3. Create a Workflow Description
  - choose from existing tools or implement new tools
4. Implement a Submission Service
  - in order to write data to your repository



# Workflow Description: Tools

**Planets IF Service Registry**

Home Registered Services Browse Local Endpoints Add External Endpoints

All

Name ↕	Registered ▲	Action
<b>GrateViewService</b> CreateView	!	
<b>JJ2000MigrateService</b> Migrate	!	
<b>JJ2000ViewerService</b> CreateView	!	
<b>JTidy</b> Migrate	!	
<b>PdfBoxMigration</b> Migrate	!	
<b>SanselanMigrate</b> Migrate	!	
<b>JhoveIdentification</b> Identify	!	
<b>JhoveValidation</b> Validate	!	
<b>MetadataExtractor</b> Characterise	!	
<b>Droid</b> Identify	✓	
<b>ImageMagickMigrate</b> Migrate	✓	
<b>SanselanIdentify</b> Identify	✓	

Info Description Register Summary

**Service Registry Information**

There are 16 Planets service endpoints deployed on this IF instance.

3 of these are recorded in this service registry.

There are 3 Planets services recorded in this Service Registry.

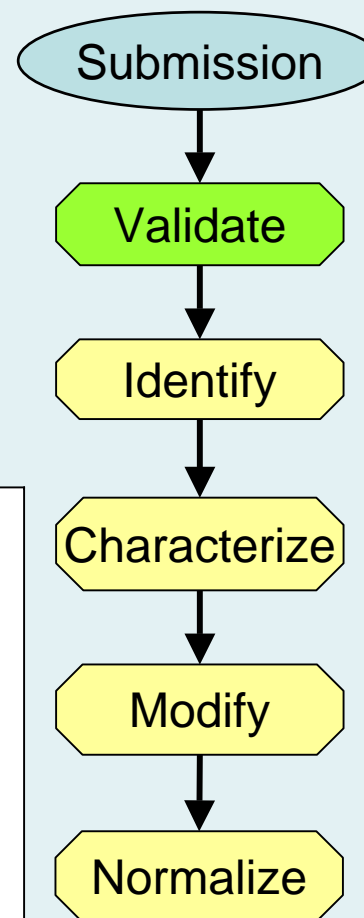
Please choose an endpoint.



# Workflow Description Step 1: Validate

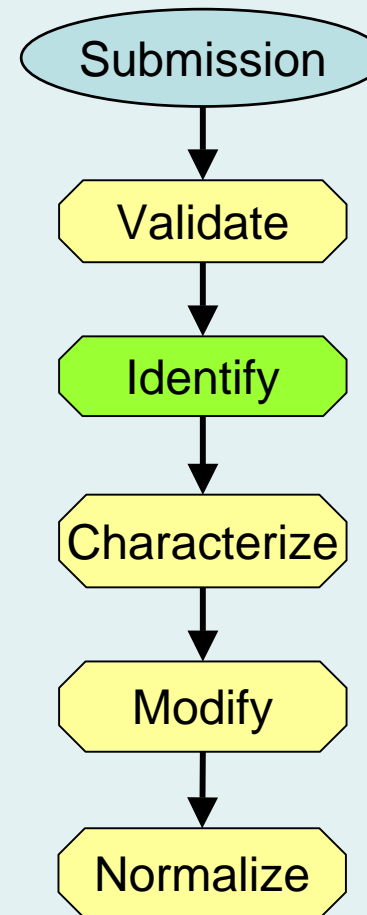
- Tool: Custom
  - check availability and validity of XML description
- Tool: JHOVE
  - Check TIFF parameters

- The file is well-formed
- The ImageLength (tag 257), ImageWidth (256), and PhotometricInterpretation (262) tags are defined
- If version 4.0 or 5.0 then StripByteCounts (279) and StripOffsets (273) are defined; if version 6.0 then either all of StripByteCounts and StripOffsets or TileByteCounts (325), TileLength (323), TileOffsets (324), and TileWidth (322) are defined
- If PhotometricInterpretation = 4, then bit 2 of NewSubFileType (254) = 1, and vice versa
- If PhotometricInterpretation = 4, then SamplesPerPixel = 1 and BitsPerSample = 1
- If PhotometricInterpretation = 0,1,3, or 4, then SamplesPerPixel = 1
- If PhotometricInterpretation = 2,6, or 8, then SamplesPerPixel = 3
- If PhotometricInterpretation = 3, then ColorMap is defined with 2BitsPerSample[0] + 2BitsPerSample[1] + 2BitsPerSample[2] values
- The values for DotRange (336) are in the range [0, (2BitsPerSample[i]-1)
- CellLength (265) defined only if Thresholding (263) = 2
- If PhotometricInterpretation = 6, then JPEGProc is defined
- If PhotometricInterpretation = 8 or 9, then BitsPerSample = 8 or 16 and SamplesPerPixel-ExtraSamples = 1 or 3
- If ClipPath (343) is defined, then XClipPathUnits (344) is defined
- TileWidth (322) and TileLength (323) values are integral multiples of 16
- DateTime (306) tag is properly formatted: "YYYY:MM:DD HH:MM:SS"



## Workflow Description Step 2: Identify

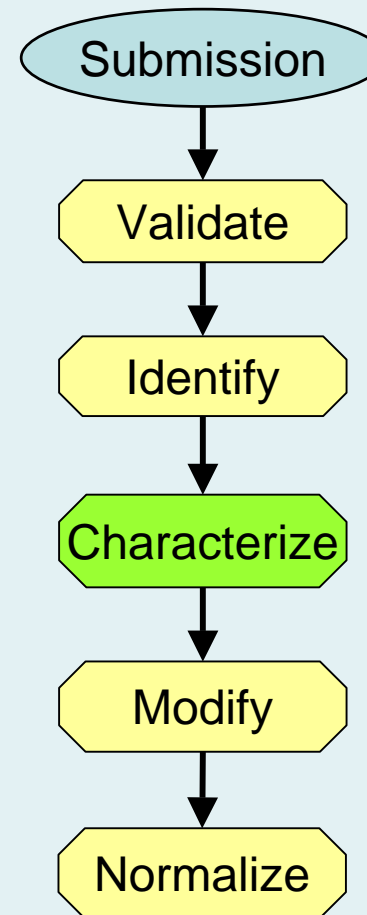
- Tool: DROID
  - PRONOM ID





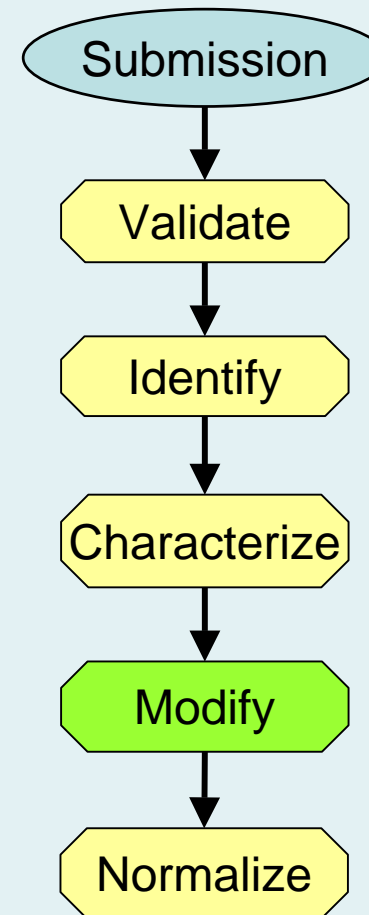
# Workflow Description Step 3: Characterize

- Tool: JHOVE
  - last modification date,  
byte size, MIME type,  
format profiles,  
checksums...



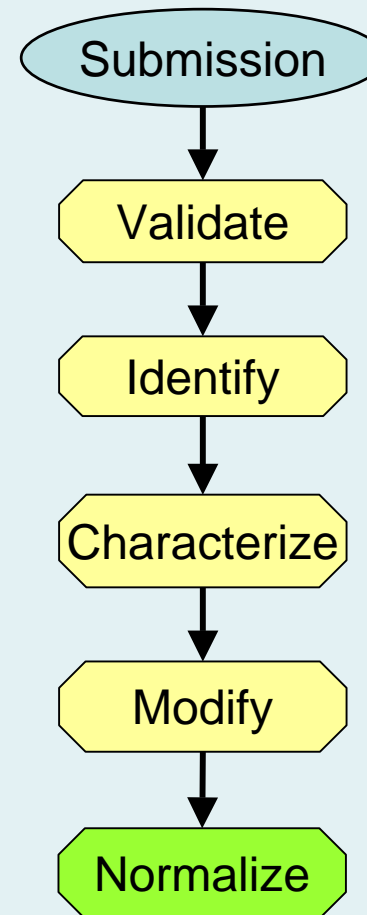
# Workflow Description Step 4: Modify

- Tool: ImageMagick
  - rotate and crop image according to metadata



# Workflow Description Step 5: Normalize

- Tool: OpenJpeg
  - convert to JPEG2000



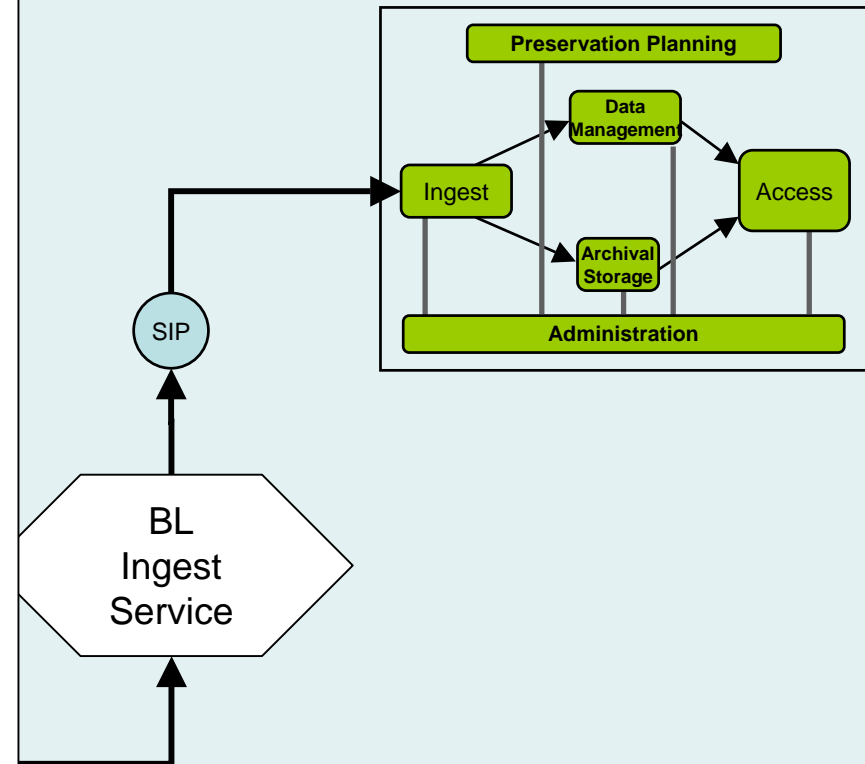
# Workflow Description

```
<?xml version="1.0" encoding="UTF-8"?>
<workflowConf xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="planets_wdt.xsd">
  <template>
    <class>eu.planets_project.ifr.core.wee.impl.templates.Submission</class>
  </template>
  <services>
    <service id="validate">
      <endpoint>http://localhost:8080/pserv-pc-jhove/JHoveValidation?wsdl</endpoint>
    </service>
    <service id="identify">
      <endpoint>http://localhost:8080/pserv-pc-droid/Droid?wsdl</endpoint>
    </service>
    <service id="characterize">
      <endpoint>http://localhost:8080/pserv-pc-jhove/JHoveIdentification?wsdl</endpoint>
    </service>
    <service id="modify">
      <endpoint>http://localhost:8080/pserv-pa-imagemagick/ImageMagicCrop?wsdl</endpoint>
      <parameters>
        <param>
          <name>boundingBox</name>
          <value>//BL_newspaper/BL_page/pageImage/pageCoordinates</value>
        </param>
        <param>
          <name>rotation</name>
          <value>//BL_newspaper/BL_page/pageImage/pageSkew</value>
        </param>
      </parameters>
    </service>
    <service id="normalize">
      <endpoint>http://localhost:8080/pserv-pa-openjpeg/OpenJpegMigrate?wsdl</endpoint>
      <parameters>
        <param>
          <name>planets:service/migration/input/migrate_to_fmt</name>
          <value>planets:fmt/ext/jp2</value>
        </param>
      </parameters>
    </service>
  </services>
</workflowConf>
```

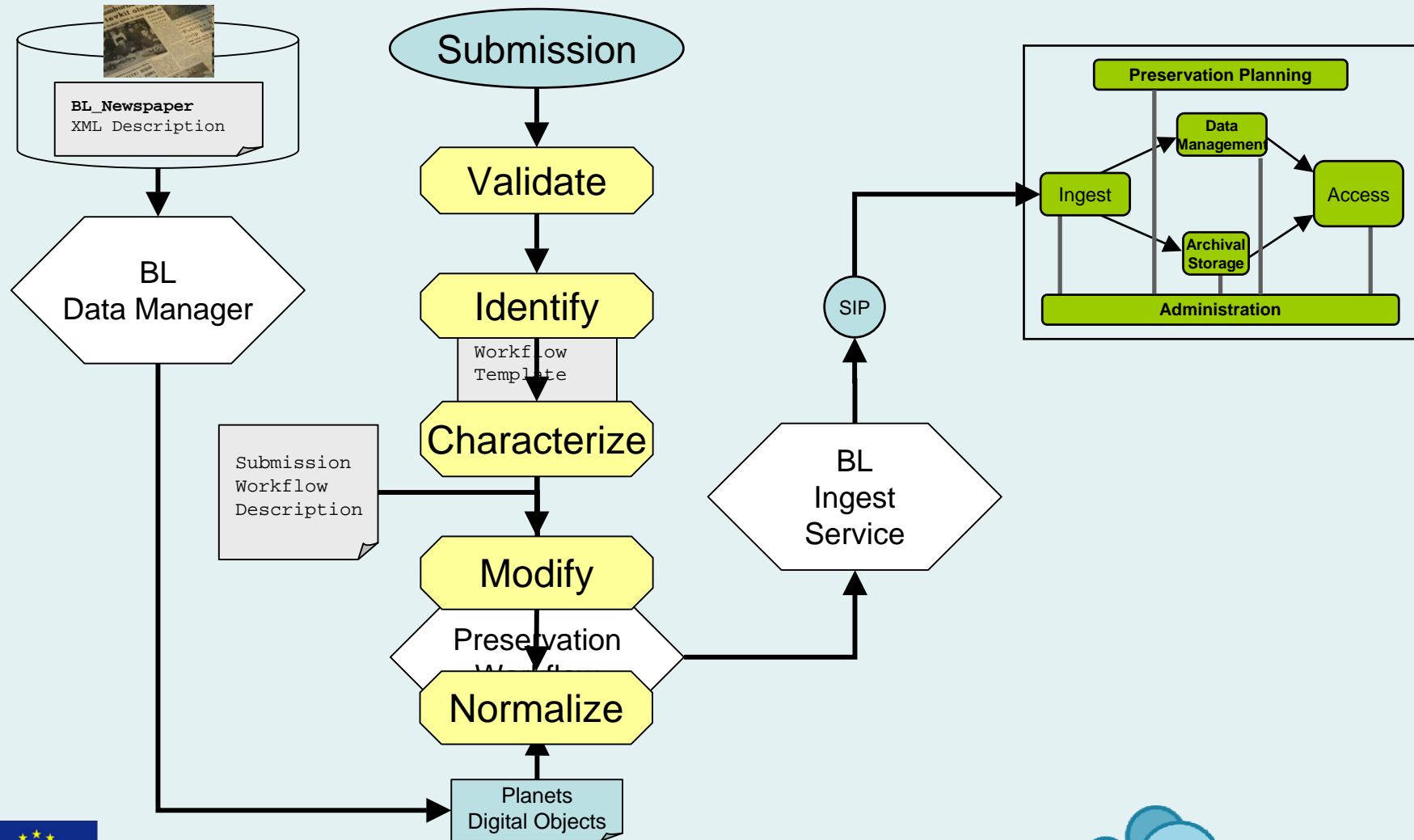


# Putting it together

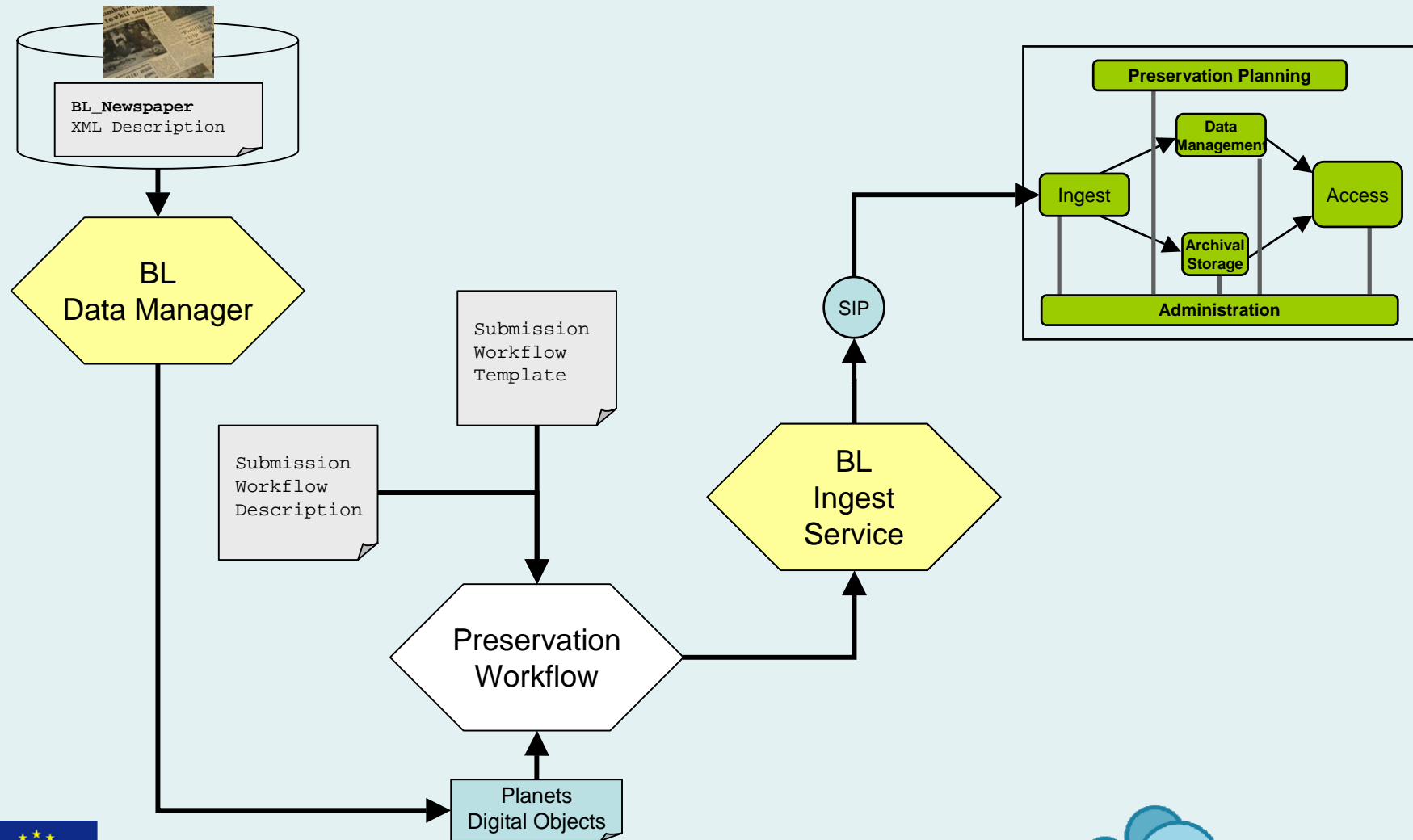
```
<?xml version="1.0" encoding="UTF-8"?>
<workflowConf xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="planets_wdt.xsd">
  <template>
    <class>eu.planets_project.ifr.core.wee.impl.templates.Submission</class>
  </template>
  <services>
    <service id="validate">
      <endpoint>http://localhost:8080/pserv-pc-jhove/JHoveIValidation?wsdl</endpoint>
    </service>
    <service id="identify">
      <endpoint>http://localhost:8080/pserv-pc-droid/Droid?wsdl</endpoint>
    </service>
    <service id="characterize">
      <endpoint>http://localhost:8080/pserv-pc-jhove/JHoveIdentification?wsdl</endpoint>
    </service>
    <service id="modify">
      <endpoint>http://localhost:8080/pserv-pa-imagemagick/ImageMagicCrop?wsdl</endpoint>
      <parameters>
        <param>
          <name>boundingBox</name>
          <value>//BL_newspaper/BL_page/pageImage/pageCoordinates</value>
        </param>
        <param>
          <name>rotation</name>
          <value>//BL_newspaper/BL_page/pageImage/pageSkew</value>
        </param>
      </parameters>
    </service>
    <service id="normalize">
      <endpoint>http://localhost:8080/pserv-pa-openjpeg/OpenJpegMigrate?wsdl</endpoint>
      <parameters>
        <param>
          <name>planets:service/migration/input/migrate_to_fmt</name>
          <value>planets:fmt/ext/jp2</value>
        </param>
      </parameters>
    </service>
  </services>
</workflowConf>
```



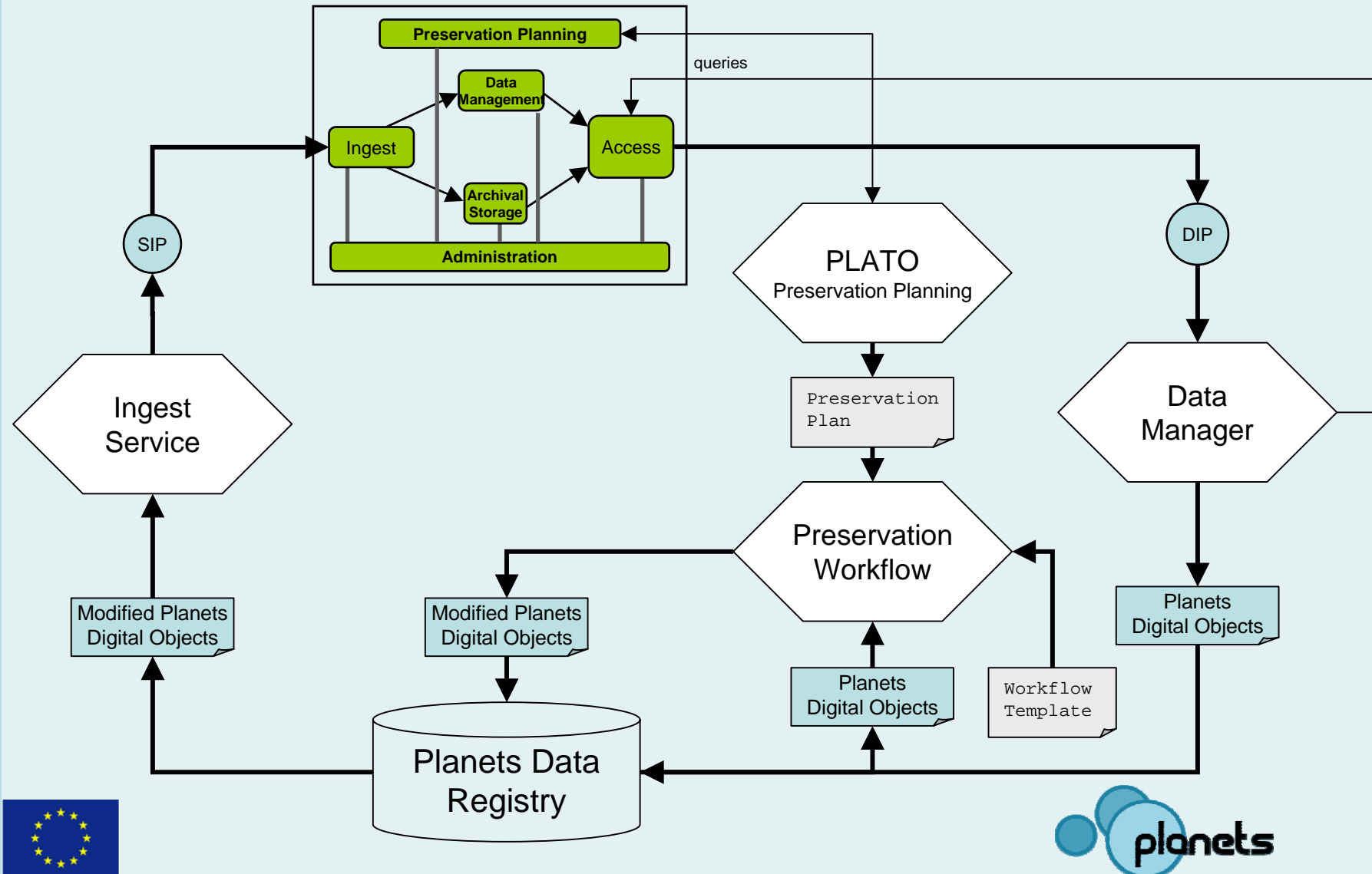
# Putting it together



# Putting it together



# Another Example: Migration





# Conclusions

---

- The OAIS model implies a number of flexible preservation workflows
- Planets Repository Integration =
  - + Data Manager
  - + Workflow Template
  - + Workflow Description/Tool Selection
  - + Ingest Service
- A number of useful preservation tools are already available
- The Planets Framework is very flexible, but requires customization



# Thank you for your attention!

---

Contact information:

Dr. Ross King

AIT Austrian Institute of Technology GmbH

[ross.king@ait.ac.at](mailto:ross.king@ait.ac.at)

