

Welcome to the History and Computing workshop

DNA - Setting up VSCodium

Step 1: Make sure you are signed in to the local Google Drive

- Press the Windows Key (start) on the keyboard and type Google Drive in the search box.
- Click on the Google Drive icon.
- Click on getting started and sign in.

Step 2: Now launch VSCodium

- Click on Windows Key (start) on the keyboard and type VSCodium in the search box.
- Click on the VSCodium icon and wait for it to open.

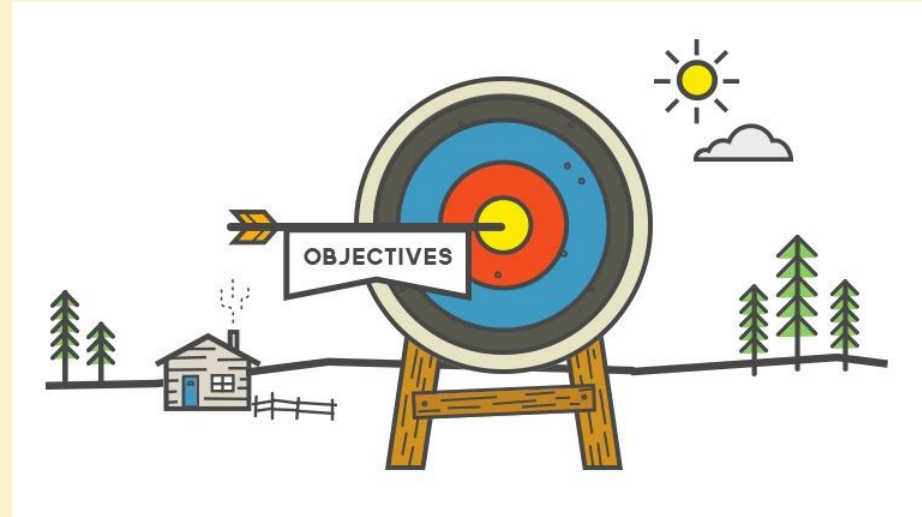
Step 3: Add the workshop folder to VSCodium Explorer

- While in VSCodium, on the top left, look for the explorer sign and click on it or (Ctrl + Shift + E). Press the 'Open Folder' button.
- Navigate to your Google Drive folder and find the folder 'history_and_computing_workshop'
- 'Select Folder' to add the 'history_and_computing_workshop' folder to the VSCodium Explorer.

History and Computing workshop

Lesson objectives - to be able to:

1. record observations as **structured data** in a digital catalogue
2. edit catalogue mark-up templates to reflect objects studied
3. encode personal stories about museum objects as **text with presentational mark-up**
4. use Quire to include catalogue data and marked-up text within a digital book
5. explain how marked-up text and catalogue data enable Quire to automate book creation



Structured data for museum objects

Museums keep structured data about their objects in **catalogues** (a specialist database)

Can you suggest some examples of the data that a museum may wish to store? What values would this data have?

Key	Example of value
Title	Anti-slavery Medallion
Creator	Genius of Universal Emancipation Newspaper
Type	Metal
Year	1830

Sharing structured data between computers

We use standard formats to move structured data from one computer system to another.

When a computer receives a file, it can follow **rules** - because the format is **standardised** - to process the data into that computer's internal representation.

Linked Art is one format for moving structured museum data between museums

...**YAML** is an alternative format

Sharing structured data between computers

Linked Art example

```
{
  "@context": "https://linked.art/ns/v1/linked-art.json",
  "id": "http://dhoxss.linkedmusic.org/AN1896-1908.AE.10.json",
  "identified_by": [
    {
      "type": "Name",
      "content": "Replica gold cup from Vapheio"
    }
  ],
  "classified_as": [
    {
      "id": "http://vocab.getty.edu/aat/300197197",
      "type": "Type",
      "_label": "Containers",
    }
  ],
  "referred_to_by": [
    {
      "type": "LinguisticObject",
      "content": "h 7.9 cm dia 10.2 cm",
      "classified_as": [
        {
          "id": "http://vocab.getty.edu/aat/300435430",
          "type": "Type",
          "_label": "Dimension Statement",
        }
      ]
    }
  ]
}
```

...

YAML example

```
- id: 1-ashmolean

  title: Replica gold cup from Vapheio

  type: Containers

  dimensions: h 7.9 cm dia 10.2 cm

  creator: Émile Gilliéron, designer

  year: late 19th - early 20th century

  accession: AN1896-1908.AE.10
```

What other rules does a computer need to know and use?

Sharing structured data between computers

YAML is a simpler format than Linked Art, more easily understood by people / grandmas / school children!

1. What would you need to go from one format to the other?
2. Can **YAML** describe everything that is in Linked Art?
3. Why do we use **YAML** then?
4. What happens to “lost data”?
5. Can we make **Linked Art** data from **YAML** data?

YAML example

```
- id: 1-ashmolean  
  
  title: Replica gold cup from Vapheio  
  
  type: Containers  
  
  dimensions: h 7.9 cm dia 10.2 cm  
  
  creator: Émile Gilliéron, designer  
  
  year: late 19th - early 20th century  
  
  accession: AN1896-1908.AE.10
```

The Concept of Key-Value Encapsulation

- **Definition:** A method to store data using pairs of keys and values
- **Efficiency:** Allows for quick data retrieval and reference
- **Usage:** Common in databases, APIs, and data serialization formats
- **Examples:** YAML and dictionaries in Python illustrate this concept

YAML example:

yaml

```
object_list:
  - id: object_1
    title: Golden Vase
    date: 1900 BCE
    description: An ancient vase made of gold.

  - id: object_2
    title: Silver Bowl
    date: 1800 CE
    description: A decorative bowl made of silver.
```

Python example:

python

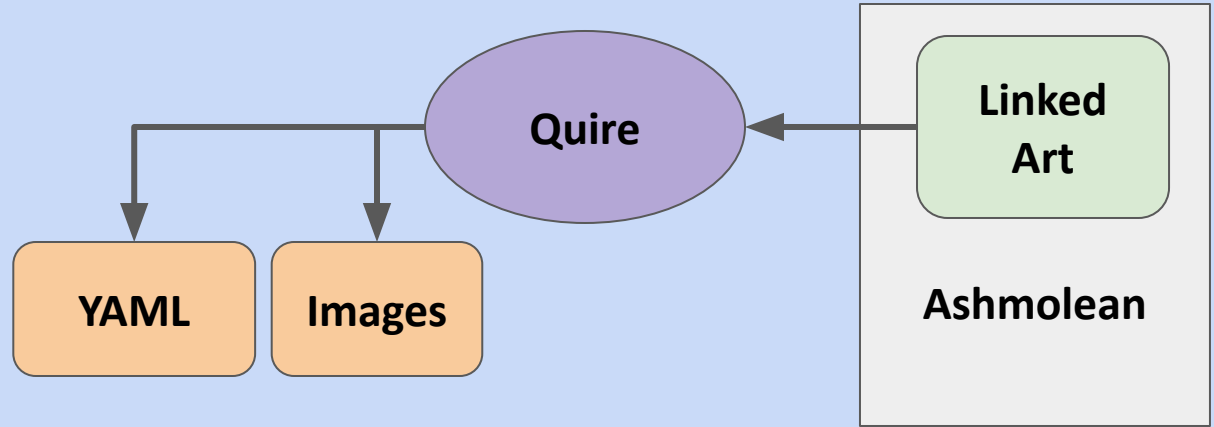
```
object_list = [
    {
        "id": "object_1",
        "title": "Golden Vase",
        "date": "1900 BCE",
        "description": "An ancient vase made of gold."
    },
    {
        "id": "object_2",
        "title": "Silver Bowl",
        "date": "1800 CE",
        "description": "A decorative bowl made of silver."
    }
]
```

Structured Data to describe our book

We're going to edit some YAML together, using key-value pairs to describe some basic properties of our book, then see what effect this has when we publish the book.

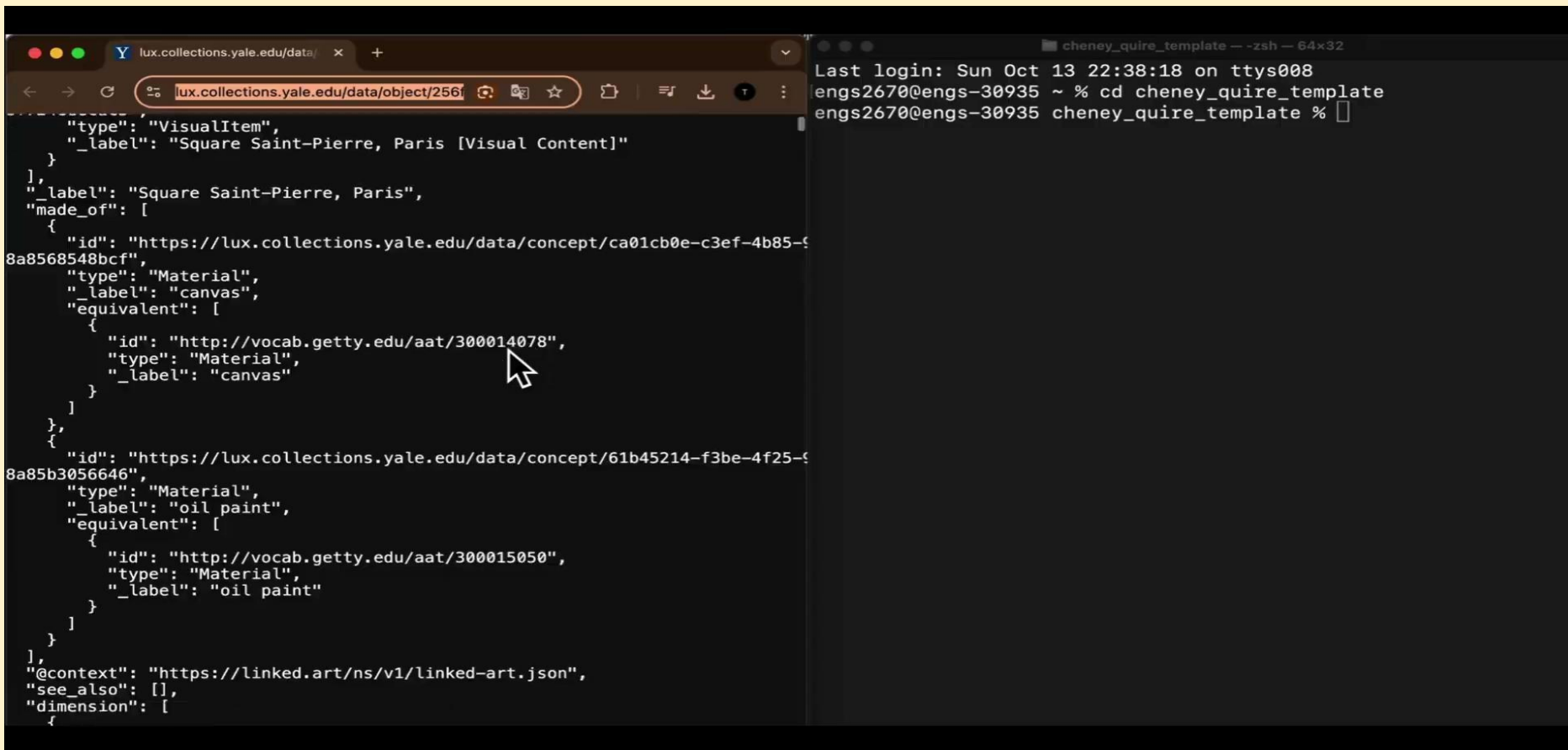
DEMO TIME!

Automated conversion of Linked Art to YAML and images



The Quire software has already been used to convert Linked Art for Ashmolean and Rumble Museum objects into YAML for you - the video you're about to see shows this. It has also downloaded images associated with that Linked Art. This is used to make the digital book.

Quire converting Linked Art to YAML



The image shows a side-by-side comparison of a web browser and a terminal window. The browser on the left displays a JSON-LD document from the Yale University Art Collection. The terminal on the right shows the command-line interface of a tool named 'cheney_quire_template', which is processing the data from the browser. The terminal output shows the last login time and the current directory, indicating the tool is ready to convert the linked art data into a YAML format.

```
lux.collections.yale.edu/data/object/256f
lux.collections.yale.edu/data/object/256f

{
  "type": "VisualItem",
  "_label": "Square Saint-Pierre, Paris [Visual Content]"
},
{
  "_label": "Square Saint-Pierre, Paris",
  "made_of": [
    {
      "id": "https://lux.collections.yale.edu/data/concept/ca01cb0e-c3ef-4b85-9
8a8568548bcf",
      "type": "Material",
      "_label": "canvas",
      "equivalent": [
        {
          "id": "http://vocab.getty.edu/aat/300014078",
          "type": "Material",
          "_label": "canvas"
        }
      ]
    },
    {
      "id": "https://lux.collections.yale.edu/data/concept/61b45214-f3be-4f25-9
8a85b3056646",
      "type": "Material",
      "_label": "oil paint",
      "equivalent": [
        {
          "id": "http://vocab.getty.edu/aat/300015050",
          "type": "Material",
          "_label": "oil paint"
        }
      ]
    }
  ]
},
{
  "@context": "https://linked.art/ns/v1/linked-art.json",
  "see_also": [],
  "dimension": [
    {

```

```
Y lux.collections.yale.edu/data/ x +
lux.collections.yale.edu/data/object/256fd9...
Pretty print ☒
{
  "id": "https://lux.collections.yale.edu/data/object/256fd9bb-5476-42fb-9071-5",
  "type": "HumanMadeObject",
  "shows": [
    {
      "id": "https://lux.collections.yale.edu/data/visual/f5f07a93-461d-49cb-95",
      "type": "VisualItem",
      "_label": "Square Saint-Pierre, Paris [Visual Content]"
    }
  ],
  "_label": "Square Saint-Pierre, Paris",
  "made_of": [
    {
      "id": "https://lux.collections.yale.edu/data/concept/ca01cb0e-c3ef-4b85-9",
      "type": "Material",
      "_label": "canvas",
      "equivalent": [
        {
          "id": "http://vocab.getty.edu/aat/300014078",
          "type": "Material",
          "_label": "canvas"
        }
      ]
    }
  ],
  {
    "id": "https://lux.collections.yale.edu/data/concept/61b45214-f3be-4f25-9",
    "type": "Material",
    "_label": "oil paint",
    "equivalent": [
      {
        "id": "http://vocab.getty.edu/aat/300015050",
        "type": "Material"
      }
    ]
  }
}
```

```
chen@quiere_template ~ -zsh - 64x32
Last login: Sun Oct 13 22:38:18 on ttys008
engs2670@engs-30935 ~ % cd cheney_quiere_template
engs2670@engs-30935 cheney_quiere_template % quire add object ht
ps://lux.collections.yale.edu/data/object/256fd9bb-5476-42fb-9071-533e3c5a4ef0
Downloading image to project's figures folder...
Linked Art added successfully. Object ID: 2. Figure ID: cat-2.
engs2670@engs-30935 cheney_quiere_template %
```

Referring to an object using an identifier

YAML example

```
- id: 1-ashmolean
```

```
  title: Replica gold cup from Vapheio
```


```
  type: Containers
```

```
  dimensions: h 7.9 cm dia 10.2 cm
```

```
  creator: Émile Gilliéron, designer
```

```
  year: late 19th - early 20th century
```

```
  accession: AN1896-1908.AE.10
```



identifier

What is the advantage of wrapping the object data in an **identifier**?

- Saves time - no need to restate the keys and values
- Reduces mistakes and errors
- Can contain **masses** of information (in other examples)

Adding your own structured object data in YAML

You now have **12 minutes** to complete the following steps on your worksheet:

Step 4. Find your group's Ashmolean object in objects.yaml

Step 5. Enter any data you recorded during the Ashmolean visit that are missing from the YAML for the object

Presentational markup in Markdown

Mark-up using Markdown syntax	Corresponding mark-up using HTML	How it's presented in a browser
<pre># Header 1 ## Header 2 ### Header 3 And so on...</pre> <p>Paragraphs are separated by a blank line.</p> <p>Text attributes: <i>*italic*</i>, **bold**, ***bold and italic***, <code>`monospace`</code>.</p>	<pre><h1>Header 1</h1> <h2>Header 2</h2> <h3>Header 3</h3> And so on...</pre> <p><p>Paragraphs are separated by a blank line.</p></p> <p><p>Text attributes: italic, bold, bold and italic, <code>monospace</code></p></p>	<p>Header 1 Header 2 Header 3</p> <p>Paragraphs are separated by a blank line.</p> <p>Text attributes: <i>italic</i>, bold, <i>bold and italic</i>, <code>monospace</code>.</p>

These are processing instructions for the computer to display your text

Presentational markup in Markdown

Mark-up using Markdown syntax	Corresponding mark-up using HTML	How it's presented in a browser
<p>Bullet lists nested within numbered list:</p> <ul style="list-style-type: none">1. fruits<ul style="list-style-type: none">* apple* banana2. vegetables<ul style="list-style-type: none">- carrot- broccoli	<pre><p>Bullet lists nested within numbered list:</p> fruits apple banana vegetables carrot broccoli </pre>	<p>Bullet lists nested within numbered list:</p> <ul style="list-style-type: none">1. fruits<ul style="list-style-type: none">● apple● banana2. vegetables<ul style="list-style-type: none">● carrot● broccoli

Markdown files have a .md extension

Presentational mark-up to introduce our book

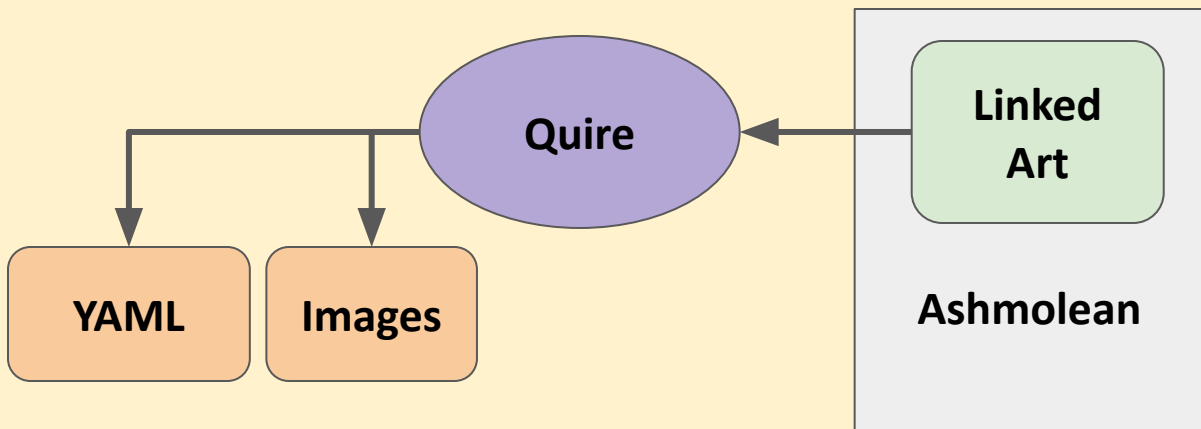
We're going to edit some Markdown together, which will be processed by Quire to create the cover page of our book.

DEMO TIME!

Adding our stories

The video showed how **Quire** can convert **Linked Art** into **YAML** and image files, to automatically include object data in our book. We then added our own observations to YAML.

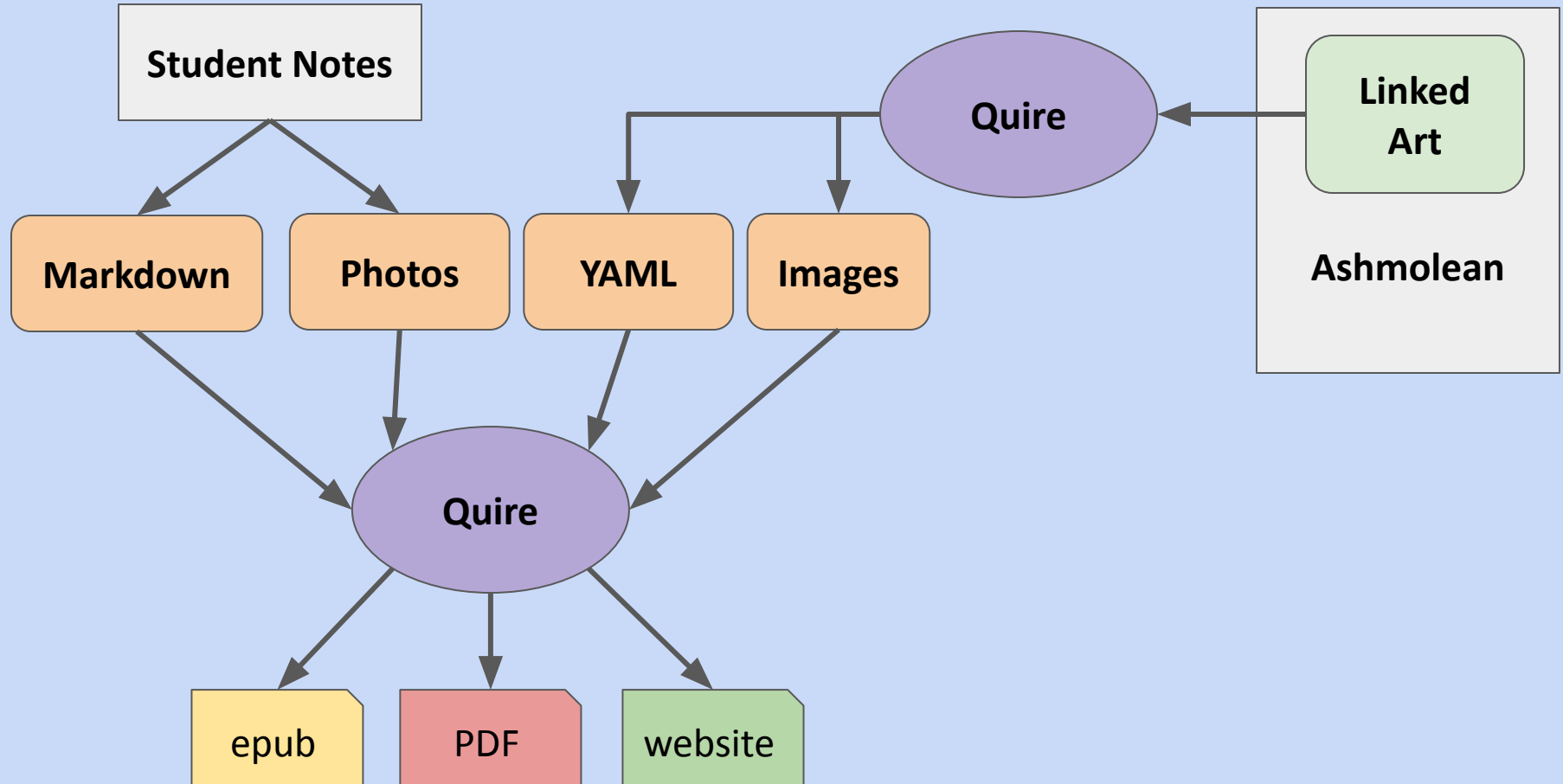
1. What other things might we wish to show or say in our book about our objects?
2. What might we want the computer to do with our written stories and interpretations?



We wish to include:

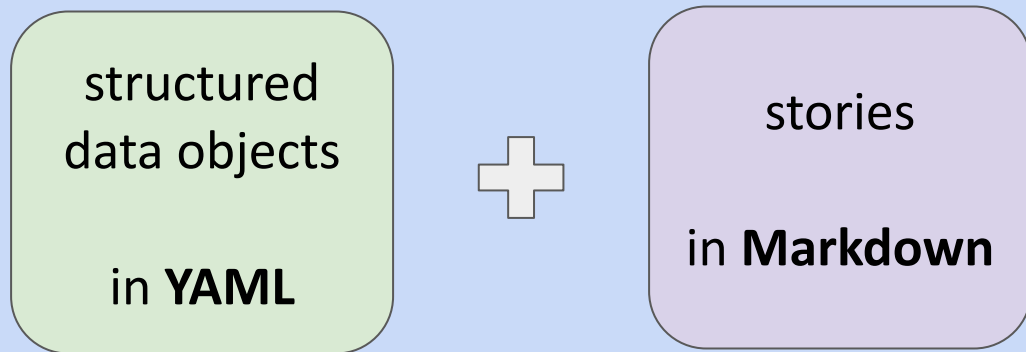
- Our photos
- Our stories
- Our text formatting e.g. bolds and italics

Combining our stories with the structured (Linked Art) data



Combining information using Quire

Quire uses two types of information processing to build our book

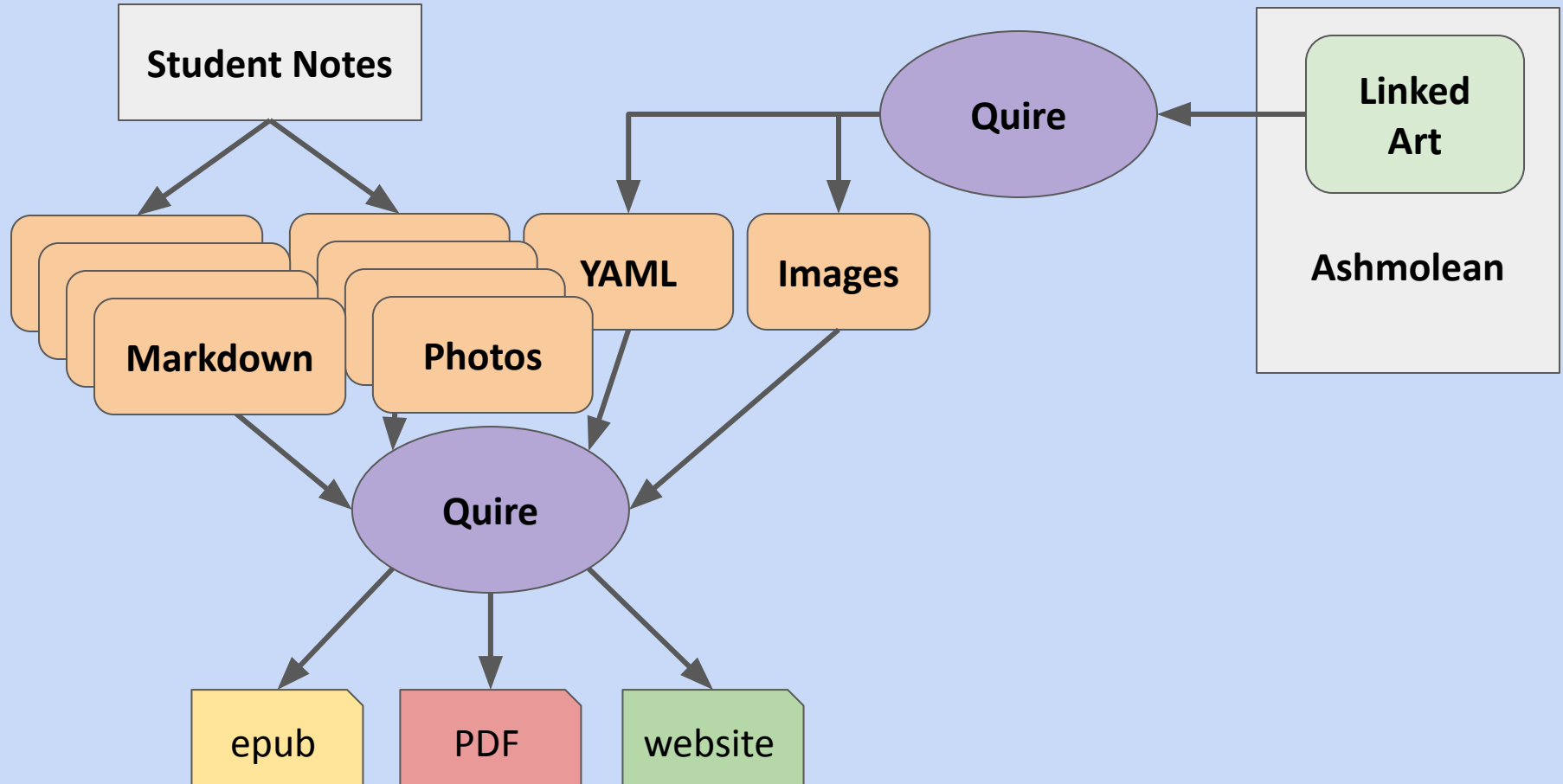


Using the **identifier** allows the story (in Markdown) to reference the structured data (in YAML)

Why do we want to use an **identifier**? How unique does the **identifier** need to be?

How could the **identifier** help with data at scale?

Combining our stories with the structured (Linked Art) data



Building the Quire Book

We're going to run three commands in Quire then check the output matches our expectations:

1. Run **quire preview** and see the preview in a web browser
2. Run **quire build**
3. Run **quire pdf** and you will check the output document

DEMO TIME!

Checking the Quire book

Whilst you have been at lunch we have been checking the Quire output...

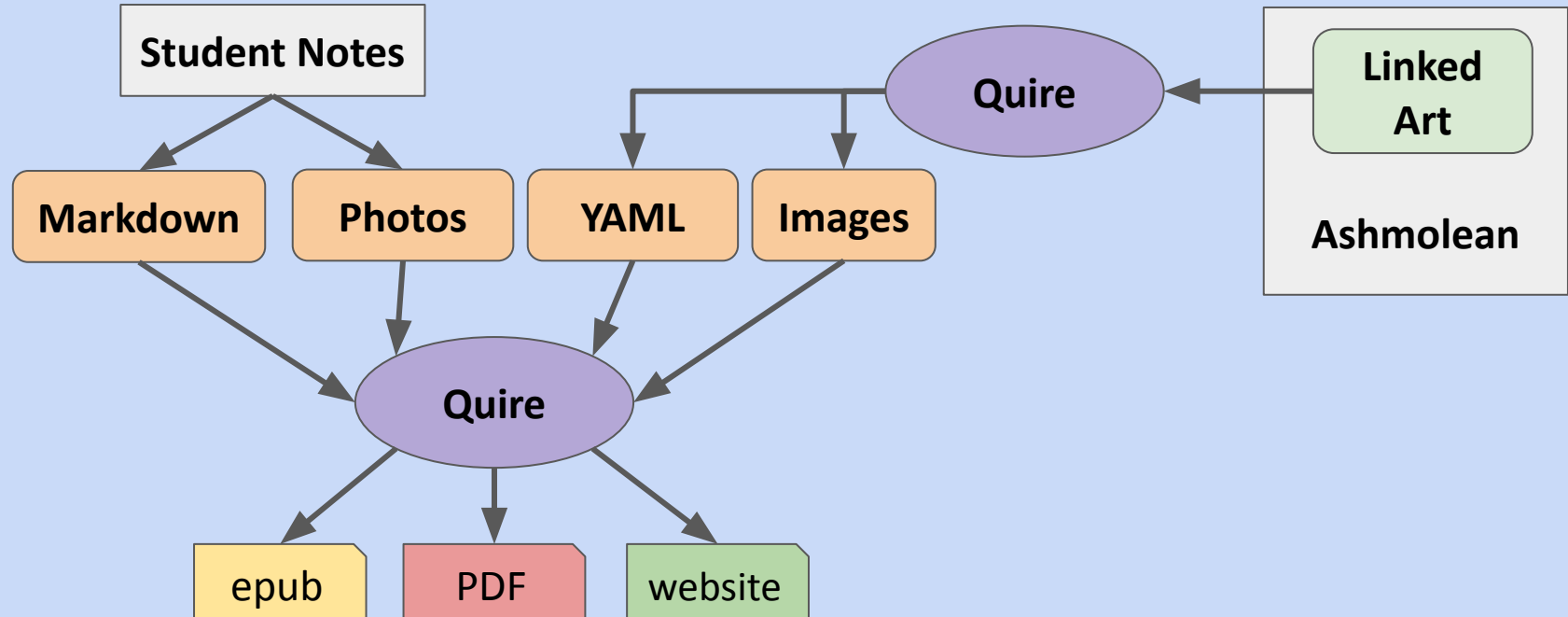
Return to your worksheet for today's session, starting from Step 9.

1. Open the shared folder 'cheney_book'. Inside this folder look for the file `pagedjs.pdf` - this is the PDF created by Quire using our book data.
2. Page through the PDF until you find the page for your group's Ashmolean object.
 - *Does it include your observations as a table? Can you see how these have been derived from the key-value pairs you entered?*
 - *Is there a picture of the object automatically included from the Ashmolean object record?*
 - *Is your story present and correct? Is your story formatted and styled as you expect?*

If any of these are missing or incorrect, let your teacher know - we may need to debug your data to uncover and correct any problems.

Final thoughts...

When you go home tonight and tell your parent about what you did today, what will you say? Can you use the diagram to help you? Write a short paragraph in a new document in the shared 'student_summaries' folder



Title?

