

デジタル・アーカイブ 周辺技術解説(1) IIF

学術コミュニケーション技術セミナー
(JPCOAR Monday)

西岡 千文(京都大学附属図書館)

2020年11月2日(月) 14:05~14:35



京都大学
KYOTO UNIVERSITY



発表内容

目的

IIIFのAPIへ理解を深めることで、IIIF対応デジタルアーカイブについての見識を得る

発表内容

- IIIF概要
- IIIFの代表的なAPIの解説
 - Image API
 - Presentation API
- IIIFの実装事例
- IIIFならではの高度な資料活用
- IIIFに関連するサイト、ソフトウェア、ツールの紹介

III F 概要

IIIF



= **International Image Interoperability Framework**

A Community
that develops **Shared APIs**,
implements them in **Software**,
and exposes interoperable **Content**

出典: NISO, "International Image Interoperability Framework (IIIF)",
<https://www.slideshare.net/BaltimoreNISO/cramer-international-image-interoperability-framework-iiif>

国際的な画像の相互利用の枠組み
共通のAPIの開発を行い、APIに対応したソフトウェアを実装し、
相互利用可能なコンテンツを公開するコミュニティ

しかし実際は「IIIF」は「IIIFのコア成果物であるAPI」を指すことが多い

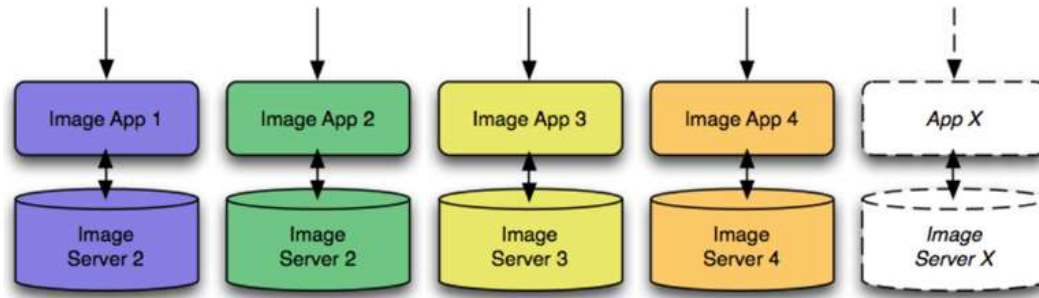
デジタルアーカイブにおける課題

過去20年以上にわたり、資料の電子化・Webでの公開が進展してきたが、下記の課題があった

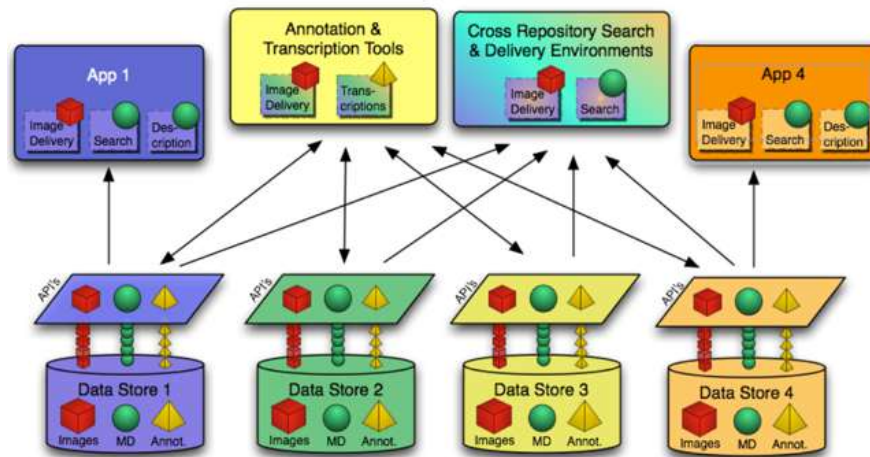
- **利用者の観点：非効率的な学術フロー**
 - 様々な機関の資料を扱う利用者は、各機関のデジタルアーカイブの操作方法を習得しなければならなかった
- **開発者の観点：開発の重複**
 - 各機関が個別に類似した機能を有するデジタルアーカイブを実装し、多大なコストを費やしてきた

IIIFは、画像と画像に付随する情報の呼び出し方法の規格をAPIとして定めることで、上記の課題を解決する

APIによる相互利用の促進



APIにより画像データと付随するメタデータの相互利用が可能に



出典: NISO, "International Image Interoperability Framework (IIIF)",
<https://www.slideshare.net/BaltimoreNISO/cramer-international-image-interoperability-framework-iiif>

III Fによる課題の解決

- **利用者の観点から: 非効率的な学術フロー**
 - 様々な機関の資料を扱う研究者は、各機関のデジタルアーカイブの操作方法を習得しなければならなかった

APIにより、一つのプラットフォーム・ソフトウェアから、あらゆる機関の画像を呼び出して閲覧することができる

- **開発者の観点から: 開発の重複**
 - 各機関で個別に類似した機能を有するデジタルアーカイブを実装し、多大なコストを費やしてきた

コミュニティでAPIに対応したフリーソフトウェアを開発・共有することで、開発が効率的となりコストが削減される

IIIFが提供するAPI

- **Image API**

- 画像の配信方法を規定

- **Presentation API**

- 資料を表示するために必要な情報の記述方法を規定

- Search API

- 資料のアノテーション等の検索の方法を規定
- 基本的にとある資料内の検索にのみ使用可能

- Authentication API

- 資料の閲覧を特定のユーザに制限するケースのために、認証プロセスの導入方法を規定

- その他、開発中のAPIが存在

- 詳細: <https://iiif.io/api/index.html>

IIIFの代表的なAPI

IIIFの代表的なAPI

- **Image API**

- 画像の配信方法を規定

- **Presentation API**

- 資料を表示するために必要な情報の記述方法を規定

- 現在(2020年10月)の最新版はバージョン3.0であるが、本資料ではバージョン2.1.1を扱う

- バージョン2.1.1と異なり、バージョン3.0では動画・音声を扱うことが可能である
- しかし、国内外の大多数の機関ではバージョン2.*を使用していることから、バージョン2.1.1を扱うこととする

Image API

(Image API 2.1.1に基づく)

Image APIの機能とURI構造

- 基底URI (Base URI)

```
{scheme}://{server}{/prefix}/{identifier}
```

例: https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00007683/RB00007683_00001_0.ptif

- Presentation API (後述) 等で使用される

- (機能1) 画像の取得

```
{scheme}://{server}{/prefix}/{identifier}/{region}/{size}  
/{rotation}/{quality}.{format}
```

例: https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00007683/RB00007683_00001_0.ptif/full/full/0/default.jpg

- (機能2) 画像情報の取得

```
{scheme}://{server}{/prefix}/{identifier}/info.json
```

例: https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00007683/RB00007683_00001_0.ptif/info.json

(機能1) 画像の取得

下記のURI構造で画像をリクエストし、取得する

```
{scheme}://{server}{/prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}
```

URIで、画像の領域、サイズ等のパラメータ(赤字箇所)を設定し、画像をリクエストする




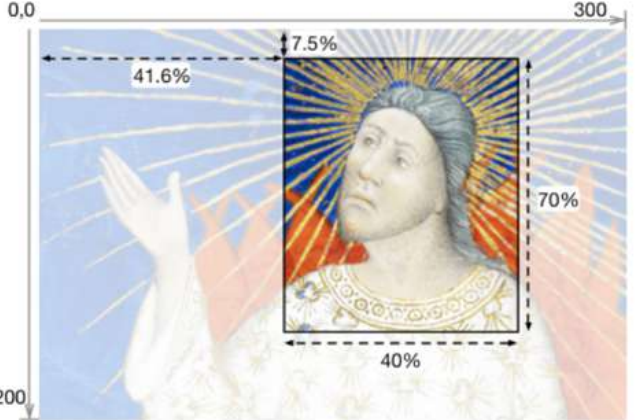
(機能1) 画像の取得: 領域

```
{scheme}://{server}/{prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}
```

- **region (領域):** リクエストする画像の領域を指定する

値	内容
full	画像の領域全体、すなわち、フル画像をリクエストする。
square	幅・高さが、フル画像の短辺の長さと同じ、正方形の画像をリクエストする。取得される領域は、画像サーバに依存するが、多くの場合、画像中央が抽出される。(例参照)
x,y,w,h	絶対ピクセル値で領域を指定する。xの値は、水平軸の0の位置からのピクセル数を表す。yの値は、垂直軸の0の位置からのピクセル数を表す。w、hは領域の幅・高さをピクセルで表す。
pct:x,y,w,h	領域を、フル画像サイズに対する%(パーセンテージ)で指定する。例えば、xは、報告されている幅の%として計算された、水平軸上の0の位置からのピクセル数を表す。wは、同じく報告されている幅の%として計算された領域の幅を表す。小数点の使用も可能である。

(機能1) 画像の取得: 領域例

 <p>0,0 300 200</p> <p>1 region=full .../full/full/0/default.jpg</p>	 <p>0,0 300 200 200</p> <p>2 region=square .../square/full/0/default.jpg</p>
 <p>0,0 300 200</p> <p>125 15 120 140</p> <p>3 region=125,15,120,140 .../125,15,120,140/full/0/default.jpg</p>	 <p>0,0 300 200</p> <p>41.6% 7.5% 40% 70%</p> <p>4 region=pct:41.6,7.5,40,70 .../pct:41.6,7.5,40,70/full/0/default.jpg</p>

(機能1) 画像の取得: サイズ [1/2]

```
{scheme}://{server}/{prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}
```

- **size (サイズ):** リクエストする画像のサイズを指定する

値	内容
full	画像または領域を、拡大・縮小せず、フルサイズでリクエストする。
max	画像または領域は、プロフィール記述のmaxWidth、maxHeight、maxAreaで示されているとおり、利用可能な最大サイズで返す。これら(maxWidth等)が指定されていない場合は、fullと同じである。
w,	画像または領域は、幅がwと等しくなるように拡大・縮小される。高さは、抽出される領域の縦横比を維持して計算された値となる。
,h	画像または領域は、幅がhと等しくなるように拡大・縮小される。幅は、抽出される領域の縦横比を維持して計算された値となる。
pct:n	画像または領域は、幅と高さのn%に拡大・縮小される。
w,h	画像の幅と高さは、wとhに指定される。結果、画像の縦横比がオリジナルから変更される場合がある。

(機能1) 画像の取得: サイズ [2/2]

前頁の表の続き

値	内容
!w,h	画像は、幅と高さが、リクエストされた幅と高さよりも小さくまたは同じになるよう、最適な大きさに拡大・縮小される。領域の縦横比を維持するように、画像サイズが算出される。

参考 : <http://www.asahi-net.or.jp/~ax2s-kmtn/ref/iiif/apiimage2.1.html#size>

(機能1) 画像の取得: サイズ例

 <p>1 size=full .../full/full/0 /default.jpg</p>	 <p>1 size=full .../full/max/0/default.jpg 画像が200ピクセルのmaxWidth を持っていると想定しているこ とに注意</p>	 <p>2 size=150, .../full/150,/0 /default.jpg</p>
 <p>3 size=,150 .../full/,150/0 /default.jpg</p>	 <p>4 size=pct:50 .../full/pct:50/0/default.jpg</p>	 <p>5 size=225,100 .../full/225,100 /0/default.jpg</p>
 <p>6 size=!225,100 .../full/!225,100 /0/default.jpg 返される画像が150,100ピ クセルであることに注意</p>		

(機能1) 画像の取得: 回転

```
{scheme}://{server}/{prefix}/{identifier}/{region}/{size}/rotation/rotation/{quality}.{format}
```

- **rotation (回転):** リクエストする画像の回転を指定する

形式	内容
n	時計回りの回転角度 (0~360) をnとして指定する。 <u>多くの画像サーバで、90の倍数以外の値には対応していない。</u>
!n	時計回りの回転角度 (0~360) をnとして指定し、画像を反転させる。

参考: <http://www.asahi-net.or.jp/~ax2s-kmttn/ref/iiif/apiimage2.1.html#rotation>

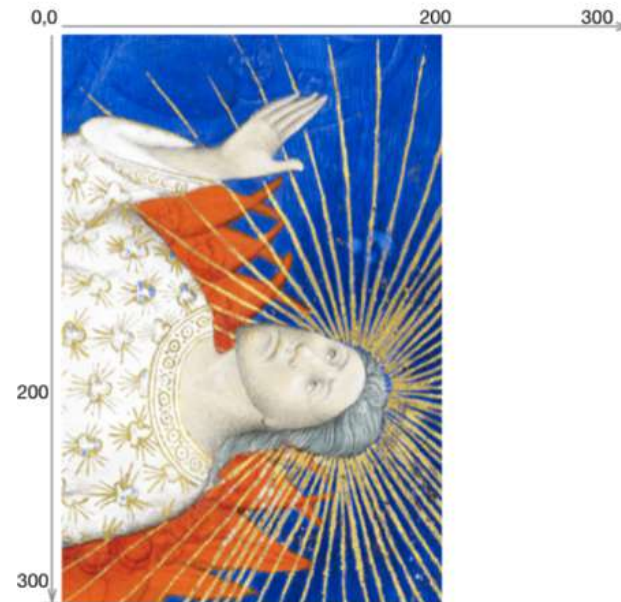
(機能1) 画像の取得: 回転例



1 rotation=0
.../full/full/0/default.jpg



5 rotation=!0
.../full/full/!0/default.jpg



3 rotation=90
.../full/full/90/default.jpg

出典: <http://www.asahi-net.or.jp/~ax2s-kmtn/ref/iiif/apiimage2.1.html#rotation>

(機能1) 画像の取得: 品質 (色調)

```
{scheme}://{server}/{prefix}/{identifier}/{region}/{size}/{rotation}/quality.{format}
```

- **quality (品質):** 画像の品質 (色調) を指定する

値	内容
color	フルカラーの画像をリクエストする。
gray	グレースケールの画像をリクエストする。
bitonal	二値画像をリクエストする。各ピクセルは、白あるいは黒である。
default	画像は、画像サーバーで設定されているデフォルト品質 (カラー、グレイ、二値) で取得される。多くの場合、デフォルトはカラーである。

参考: <http://www.asahi-net.or.jp/~ax2s-kmtn/ref/iiif/apiimage2.1.html#quality>

(機能1) 画像の取得: 品質(色調)例



出典: <http://www.asahi-net.or.jp/~ax2s-kmtn/ref/iiif/apiimage2.1.html#quality>

(機能1) 画像の取得: フォーマット

```
{scheme}://{server}/{prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}
```

- **format (フォーマット):** 画像のフォーマットを指定する

値	内容
jpg	jpg
tif	tif
png	png
gif	gif
jp2	JPEG2000
pdf	PDF
webp	WebP (ウェッピー): トラフィック量軽減と表示速度短縮を目的として、Googleにより開発されている。

参考: <http://www.asahi-net.or.jp/~ax2s-kmtn/ref/iiif/apiimage2.1.html#format>

画像配信の仕組み

- IIF対応の画像サーバでは、複数種類のサイズの画像を用意しておき、リクエストに応じて適切なサイズの画像を配信する
- 一定以上大きな画像については分割(タイル画像化)して必要なタイルのみを配信するという仕組みが用意されている
- 上記の仕組みによって、IIFは高精細画像の配信を可能としている
 - 例えば、ビューワで画像を拡大・縮小するたびに、リクエストされた領域やサイズに応じて、適したタイルがImage APIを通じて呼び出されている

参考: 永崎研宣. (2018). [C14] 国際的な画像共有の枠組み IIF の課題と展望. デジタルアーカイブ学会誌, 2(2), 111-114.

(機能2) 画像情報の取得

```
{scheme}://{server}/{prefix}/{identifier}/info.json
```

例: https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00007683/RB00007683_00001_0.ptif/info.json

JSON-LD形式で画像の情報が配信される

<pre>{ "@context" : "http://iiif.io/api/image/2/context.json", "@id" : "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00007683/RB00007683_00001_0.ptif", "protocol" : "http://iiif.io/api/image", "width" : 6464, "height" : 5880, "sizes" : [{ "width" : 50, "height" : 45 }, { "width" : 101, "height" : 91 }, { "width" : 202, "height" : 183 }, { "width" : 404, "height" : 367 }, { "width" : 808, "height" : 735 }, { "width" : 1616, "height" : 1470 }], "tiles" : [{ "width" : 256, "height" : 256, "scaleFactors" : [1, 2, 4, 8, 16, 32, 64, 128] }], "profile" : ["http://iiif.io/api/image/2/level1.json", { "formats" : ["jpg"], "qualities" : ["native","color","gray"], "supports" : ["regionByPct","sizeByForcedWh","sizeByWh","sizeAboveFull","rotationBy90s","mirroring","gray"] }] }</pre>	基本情報
	画像のサイズ
	画像サーバが提供する複数種類のサイズ
	タイル情報
	適用可能な機能等

Image APIへの対応

- 画像をImage APIに対応させるには、Image APIに対応する画像サーバに画像を保存する必要がある
- 主な画像サーバ
 - IIP Imageサーバ
 - <https://iipimage.sourceforge.io/documentation/server/>
 - 高性能
 - 多くのデジタルアーカイブで利用されている
 - Loris
 - <https://github.com/loris-imageserver/>
 - Cantaloupe
 - <https://cantaloupe-project.github.io/>



Presentation API

(Presentation API 2.1.1に基づく)

Presentation APIの機能

- Presentation APIは、資料をビューワで表示するために必要な情報を記述する方法(データモデル)を規定している
 - 資料をビューワで表示するために必要な情報としては、「どの資料のどの頁にどの画像が含まれるか」「ビューワで表示する資料のタイトル等のメタデータ」が挙げられる
- 情報はJSON-LD形式(補足参照)で記述する
- あくまでも表示のためのAPIであり、メタデータの記述のためではない
- 実物とPresentation APIにおけるオブジェクトの対応
 - コレクション≡コレクション(Collection)
 - 資料≡マニフェスト(Manifest)
 - 頁≡キャンバス(Canvas)

マニフェスト

- Presentation APIでは、各資料はマニフェストというJSONオブジェクトに対応している
- 多くのデジタルアーカイブで、マニフェストはマニフェストファイル(manifest.json)として公開されている
- ビューワはマニフェストファイルを読み込み、資料を表示する

マニフェストファイルに含まれている情報

- ファイルで表現されているJSONオブジェクトがマニフェストであることを示す情報
- 資料全体に関する情報(資料タイトル、権利情報等)
- 資料の構成に関する情報

```

"@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00020007/manifest.json",
"@type": "sc:Manifest",
"@context": "http://iiif.io/api/presentation/2/context.json",
"label": "山城国[畿内図]",
"metadata": [
  {
    "label": "タイトル",
    "value": "山城国[畿内図]"
  }
],
"description": "This is...",
"attribution": "Kyoto University Rare Materials Digital Archive",
"license": "https://rmda.kulib.kyoto-u.ac.jp/reuse",
"logo": {
  "@id": "https://rmda.kulib.kyoto-u.ac.jp/themes/custom/kyoto_dc/images/logo_u.png"
},
"viewingDirection": "left-to-right",
"within": "https://rmda.kulib.kyoto-u.ac.jp/collection/nakai".

```

IIIF マニフェストであることを示す情報

資料オブジェクト全体に関する情報
(資料タイトル、権利情報等)

```

"sequences": [
  {
    "@type": "sc:Sequence",
    "canvases": [
      {
        "label": " ",
        "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00020007/page173025",
        "@type": "sc:Canvas",
        "height": 2933,
        "width": 3800,
        "images": [
          {
            "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00020007/ano173025",
            "@type": "oa:Annotation",
            "motivation": "sc:painting",
            "on": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00020007/page173025",
            "resource": {
              "description": "[paragraph:field_description]",
              "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00020007/img173025",
              "@type": "dctypes:Image",
              "format": "image/jpeg",
              "service": {
                "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00020007/RB00020007_00001_0.ptif",
                "@context": "http://iiif.io/api/image/2/context.json",
                "profile": "http://iiif.io/api/image/2/level1.json"
              }
            }
          }
        ]
      }
    ]
  }
]

```

資料の構成に関する情報

IIIF マニフェスト例

マニフェストであることを示す情報

プロパティ名	内容
<u>@context</u>	マニフェストファイル (JSON-LD) のコンテキストのURI 例: http://iiif.io/api/presentation/2/context.json
<u>@id</u>	マニフェストのID (多くの場合、マニフェストファイルのURI) 例: https://rmda.kulib.kyoto-u.ac.jp/iiif/metadata_manifest/RB00000125/manifest.json
<u>@type</u>	マニフェストを示す型で常に "sc:Manifest" ファイルがIIIFマニフェストであることを示すために必要

全て必須項目

参考: 神崎正英. 画像共有の新しい標準IIIF. LOD Diary.
<https://service.infocom.co.jp/das/loddiary/2017/01/20170127001583.html>

資料全体に関する情報

プロパティ名	内容
<u>label</u>	資料のタイトル
<u>description</u>	資料の概要説明(テキスト)
<u>metadata</u>	任意のメタデータを、labelとvalueで記述したJSONオブジェクトの配列
attribution	資料の権利者、提供者等
license	資料のライセンス情報への外部リンク 例: https://rmda.kulib.kyoto-u.ac.jp/reuse
viewingDirection	頁めくり方向を指定(デフォルト: left-to-right) 日本語資料の多くは"right-to-left"が適切
seeAlso	資源情報をセマンティックに記述している機械可読ドキュメント(XMLやRDF)へのリンク ドキュメントは、検索・発見、または資料に関する詳細な情報を提供するために使用

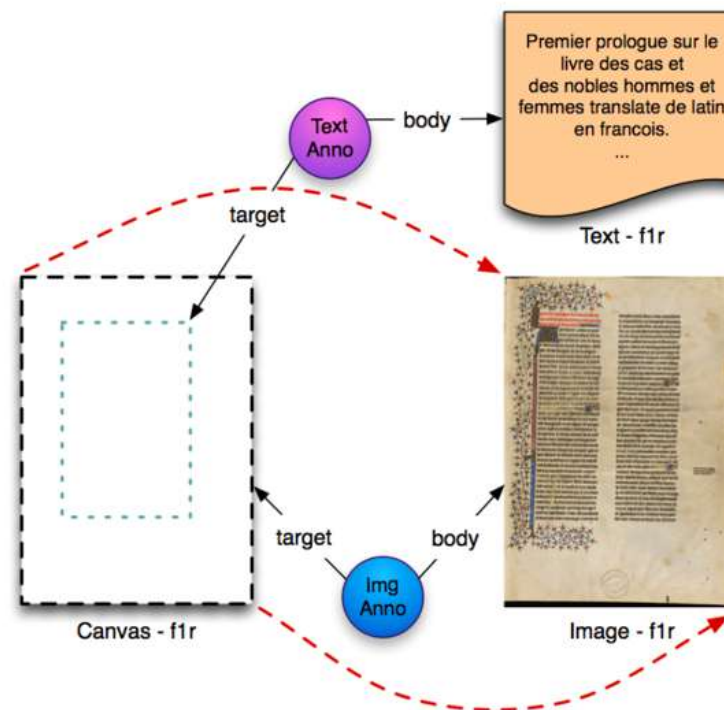
labelは必須、descriptionとmetadataは推奨

参考: 神崎正英. 画像共有の新しい標準IIIF. LOD Diary.
<https://service.infocom.co.jp/das/loddiary/2017/01/20170127001583.html>

資料の構成に関する情報： 背景にあるデータモデル [1/2]

資料の構成に関する情報の記述方法の背景にある データモデル：**Shared Canvas**

古い写本は、頁の一部が抜けていたり、散逸して複数機関で保有されていたりするため、抽象的な空白ページ(共有キャンバス)を用意し、画像やテキストなどを表示して、共有化



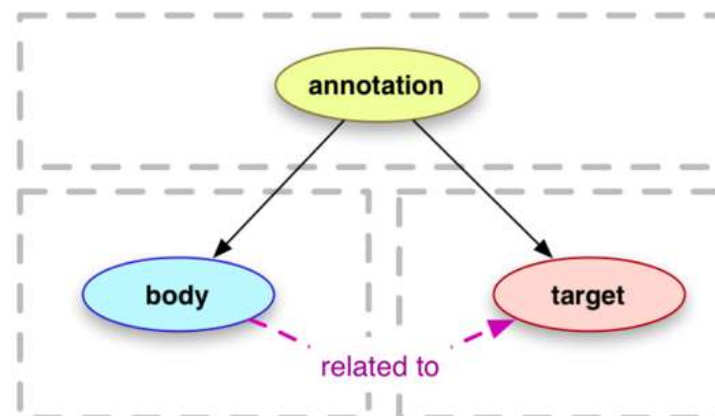
出典：Sanderson, R., Albritton, B., Schwemmer, R., & Van de Sompel, H. (2011, June). Sharedcanvas: a collaborative model for medieval manuscript layout dissemination. In *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries* (pp. 175-184).

参考：神崎正英. IIIIFマニフェストと注釈. LOD Diary.

<https://service.infocom.co.jp/das/loddiary/2017/04/20170410001609.html>

資料の構成に関する情報： 背景にあるデータモデル [2/2]

- Web上の注釈を共有・連携させるためのモデルであり、Shared Canvasはこのモデルに基づいている
- 共有・連携のため、Linked Dataの考え方を導入している
- annotationはtargetとbodyにより構成されている
 - targetがキャンバス(空白ページ)、bodyが画像やテキスト
- Web Annotation Data Modelとして2017年2月にW3C推奨
→Presentation APIのデータモデルはWWWの標準に準拠している



資料の構成に関する情報:リソースの種類 [1/2]

資料の構成に関する情報=資料の設計図

以下は、資料を構成するリソース(ブロックや煉瓦のようなもの)

- シーケンス
 - キャンバスの順序つきリスト→頁の順序についての情報
- キャンバス
 - 頁を表現する仮想的な入れ物である
 - 頁または頁の一部と関連するコンテンツを有する
 - 空白キャンバスが表示され、その上にコンテンツ(画像、テキスト)をアノテーションとして付与・表示する
- コンテンツ
 - 頁に関連付けられる画像や注釈等のテキスト
 - 本資料では、「画像」のみを扱う

資料の構成に関する情報:リソースの種類 [2/2]

マニフェスト(≡資料)

シーケンス1

キャンバス1(≡頁)

画像1

(コンテンツ)

キャンバス2(≡頁)

画像1

(コンテンツ)

...

キャンバス10(≡頁)

画像1

(コンテンツ)

シーケンス2

キャンバス1(≡頁)

画像1

(コンテンツ)

画像2

(コンテンツ)

...

キャンバス3(≡頁)

画像1

(コンテンツ)

```

"sequences": [
  {
    "@type": "sc:Sequence",
    "canvases": [
      {
        "label": " ",
        "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/page139350",
        "@type": "sc:Canvas",
        "height": 2933,
        "width": 3800,
        "images": [
          {
            "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/ano139350",
            "@type": "oa:Annotation",
            "motivation": "sc:painting",
            "on": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/page139350",
            "resource": {
              "description": "[paragraph:field_description]",
              "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/img139350",
              "@type": "dctypes:Image",
              "format": "image/jpeg",
              "service": {
                "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/RB00013739_00001_0.ptif",
                "@context": "http://iiif.io/api/image/2/context.json",
                "profile": "http://iiif.io/api/image/2/level1.json"
              }
            }
          }
        ]
      }
    ]
  },
  {
    "label": " ",
    "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/page139351",
    "@type": "sc:Canvas",
    "height": 2933,
    "width": 3800,
    "images": [
      {
        "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/ano139351",
        "@type": "oa:Annotation",
        "motivation": "sc:painting",
        "on": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/page139351",
        "resource": {
          "description": "[paragraph:field_description]",
          "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/img139351",
          "@type": "dctypes:Image",
          "format": "image/jpeg",
          "service": {
            "@id": "https://rmda.kulib.kyoto-u.ac.jp/iiif/RB00013739/RB00013739_00002_0.ptif",
            "@context": "http://iiif.io/api/image/2/context.json",
            "profile": "http://iiif.io/api/image/2/level1.json"
          }
        }
      }
    ]
  }
]

```

① シーケンスの箱 (配列)

② 一つ目のシーケンス

③ キャンバスの箱 (配列)

④ 一枚目、二枚目のキャンバス

⑤ キャンバスに貼り付ける画像 (コンテンツ) の箱 (配列)

⑥ キャンバスに貼り付ける一枚目の画像

資料の構成に関する情報:シーケンス

• シーケンス

プロパティ名	内容
<u>sequences</u>	マニフェストが持つシーケンスの配列 (p. 39の①)

- 1つの資料につき複数の並び順序を与えることも可能

• シーケンスの記述 (p. 39の②)

プロパティ名	内容
<u>@type</u>	シーケンスを示す型で常に”sc:Sequence”
<u>canvases</u>	カンバスを表すJSONオブジェクトの配列 (p. 39の③)

- カンバスは資料の個々の頁に相当

参考: 神崎正英. 画像共有の新しい標準IIIF. LOD Diary.
<https://service.infocom.co.jp/das/loddiary/2017/01/20170127001583.html>

資料の構成に関する情報:カンバス

- カンバスの記述 (p. 39の④)
 - canvasesのリスト要素がもつプロパティ

プロパティ名	内容
<u>@type</u>	カンバスを示す型で常に”sc:Canvas”
<u>@id</u>	カンバスを識別するID (URI)
<u>height</u>	カンバスの高さのピクセル値
<u>width</u>	カンバスの幅のピクセル値
images	表示する画像のリスト (p. 39の⑤)

参考: 神崎正英. 画像共有の新しい標準IIIF. LOD Diary.
<https://service.infocom.co.jp/das/loddiary/2017/01/20170127001583.html>

資料の構成に関する情報: 画像

! この辺りの事項はPresentation API 3.0で変更!

- 画像の記述 (p. 39の⑥)
 - imagesのリスト要素がもつプロパティ

プロパティ名	内容
<u>@type</u>	キャンバスに対する注釈を示す型で常に”oa:Annotation”
<u>motivation</u>	注釈の種類を表現し、常にsc:painting
<u>resource</u>	注釈で表現する画像リソース
<u>on</u>	注釈が含まれるキャンバス(キャンバスの@idの値と等しい) フラグメントにより、キャンバスのどの位置に画像を配置するか指定することが可能 例: http://example.org/iiif/book1/canvas/p1#xywh=100,100,500,300

参考: 神崎正英. 画像共有の新しい標準IIIF. LOD Diary.
<https://service.infocom.co.jp/das/loddiary/2017/01/20170127001583.html>

資料の構成に関する情報: 画像リソース・IIIFサービス

! この辺りの事項はPresentation API 3.0で変更!

• 画像リソース

プロパティ名	内容
<u>@id</u>	画像リソースのURI
<u>@type</u>	画像リソースを示す型であり”dctypes:Image”が望ましい
<u>service</u>	画像リソースを提供するIIIFサービス(下記参照)

• IIIFサービス

プロパティ名	内容
<u>@context</u>	画像APIのコンテキストのURI 例: http://iiif.io/api/image/2/context.json
<u>@id</u>	画像の基底URI
service	Image APIの機能レベル

参考: 神崎正英. 画像共有の新しい標準IIIF. LOD Diary.
<https://service.infocom.co.jp/das/loddiary/2017/01/20170127001583.html>

資料の構成に関する情報: 扱わなかった内容

- 翻刻・翻訳等の注釈
 - 画像と同じ要領で、テキストをアノテーションとしてキャンバスに付与
 - 細かい規定については、Presentation APIを確認
- 目次など資料の内容の構成についての表現
 - プロパティ名: structures
 - structuresの値は、レンジ(sc:Range)の配列
 - レンジは章に相当

マニフェストファイルを読み込み、 資料を閲覧するには？ [1/3]

IIIF対応のビューワで、マニフェストファイルを読み込む必要がある。

代表的なIIIF画像ビューワ

- Mirador
 - 多くの機能が実装されている
 - 特に、研究者向けの機能が充実しており、画像の比較ができる（後述）
 - 日本語資料に適した頁方向”right-to-left”に対応
- Universal Viewer (UV)
 - 比較的シンプル
 - 動画・音声などのあらゆるメディアデータに対応
 - 画像のダウンロードボタンを提供
 - 日本語資料に適した頁方向”right-to-left”に対応

Mirador

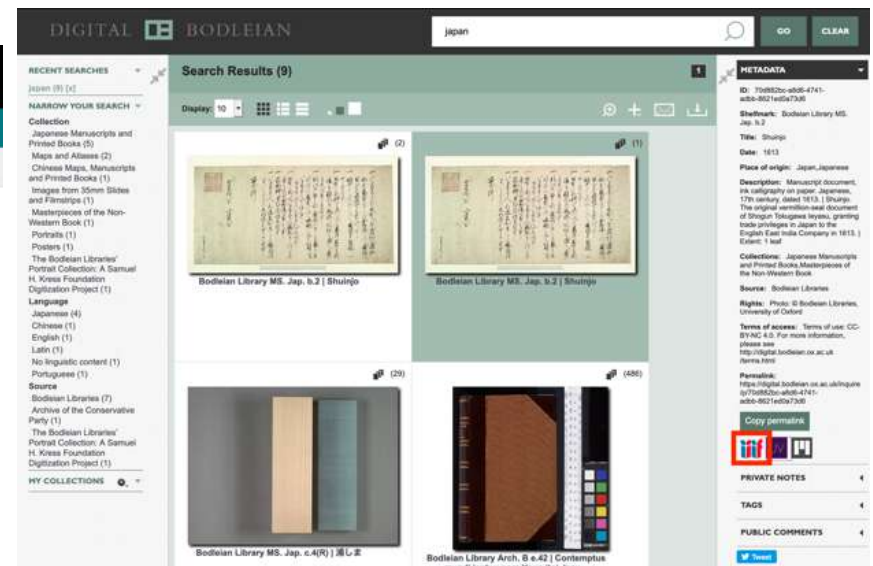
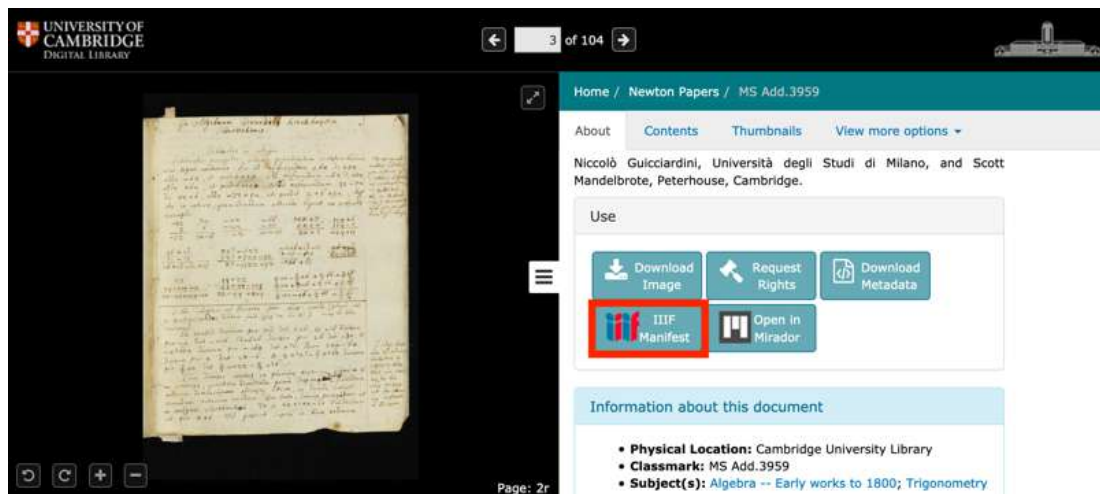


『甲斐徳本翁抄書』(京都大学附属図書館所蔵)

マニフェストファイルを読み込み、 資料を閲覧するには？ [2/3]


マニフェストファイルの読込方法

- Mirador: IIFマニフェストアイコンのドラッグ&ドロップ
 - IIFマニフェストアイコンはマニフェストファイルへのリンクを提供するアイコンであり、多くのデジタルアーカイブに表示されている



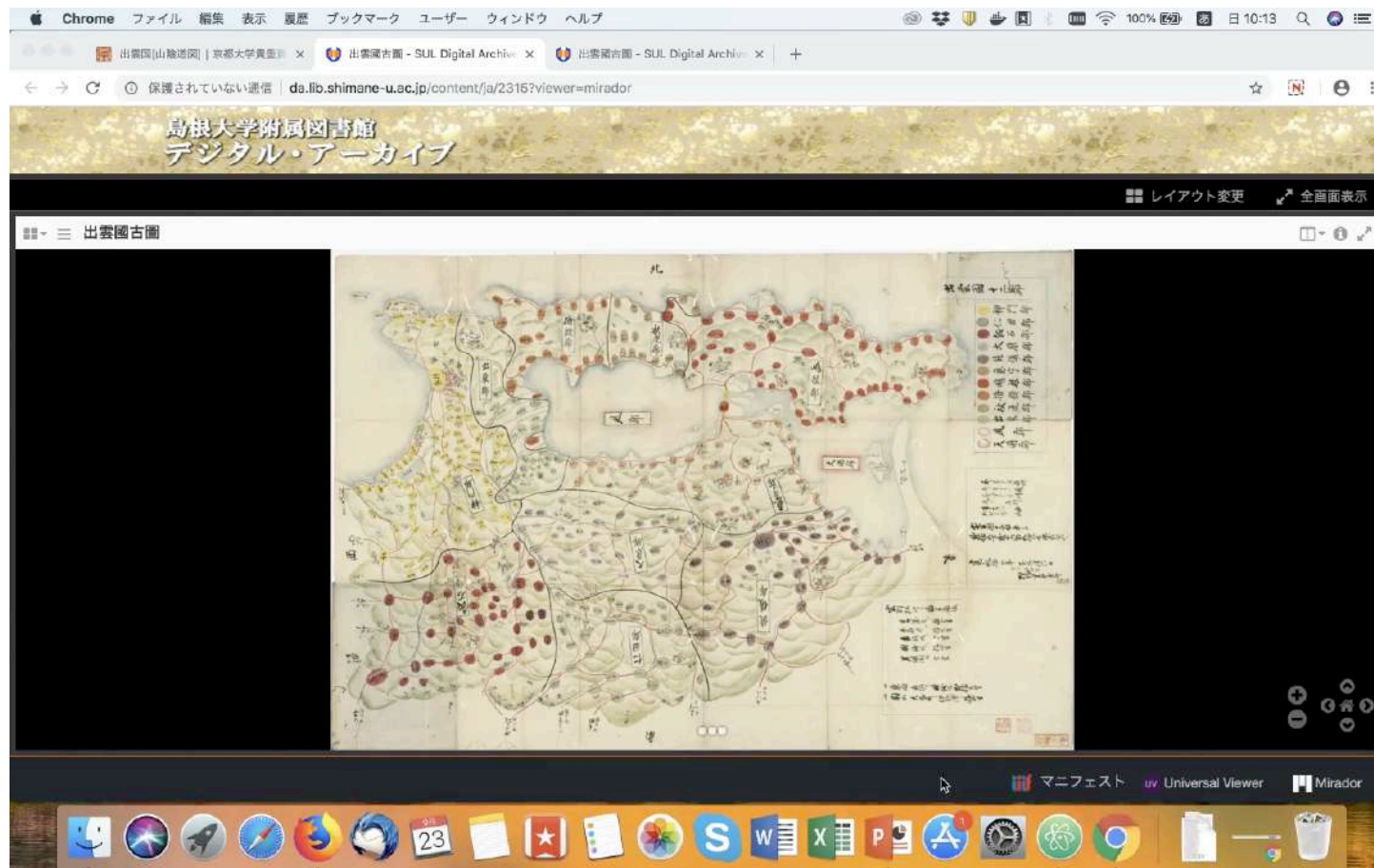
- ドラッグ&ドロップの操作については48、49頁参照

マニフェストファイルを読み込み、 資料を閲覧するには？ [3/3]

- Mirador、UV: マニフェストファイルのURIを使用した方法
 - IIIFマニフェストアイコンよりマニフェストファイルのURIを取得する
 - Miradorの場合
 - ビューワの左上にあるアイコン  をクリックし、「新しいオブジェクト」を選択
 - 「URLで新規オブジェクトを追加」の横のテキストボックスに、マニフェストファイルのURLを入力
 - UVの場合
 - 下記のURLにマニフェストファイルのURLを入力
`https://universalviewer.io/uv.html?manifest=[manifest url]`

MiradorでのIIIFマニフェストアイコンのドラッグ&ドロップと資料の比較 [1/2]

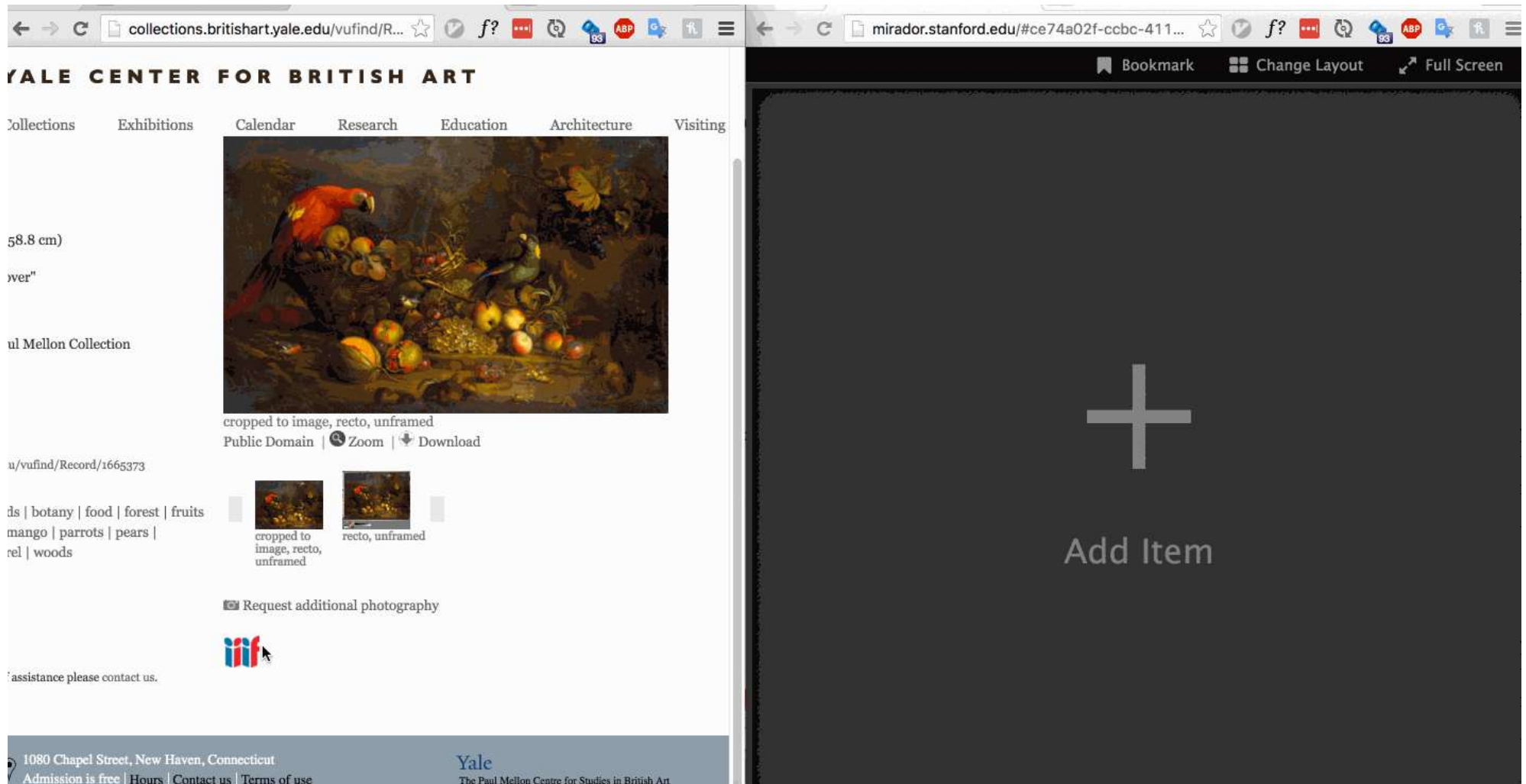
IIIFマニフェストアイコンは、IIIFマニフェストファイル(資料をビューワに表示するための情報を保有)へのリンクを提供する。Mirador等のビューワにアイコンをドラッグ&ドロップすることで、資料を読み込ませることが容易にできる。



下記の動画の5:00
~6:25を参照
<https://youtu.be/7X4gLUoCc9Y>

ユーザは一つのビューワで様々な機関の資料を閲覧することができる

MiradorでのIIIFマニフェストアイコンのドラッグ&ドロップと資料の比較 [2/2]



出典: https://iiif.io/explainers/using_iiif_resources

マニフェストファイルの作成方法 [1/2]

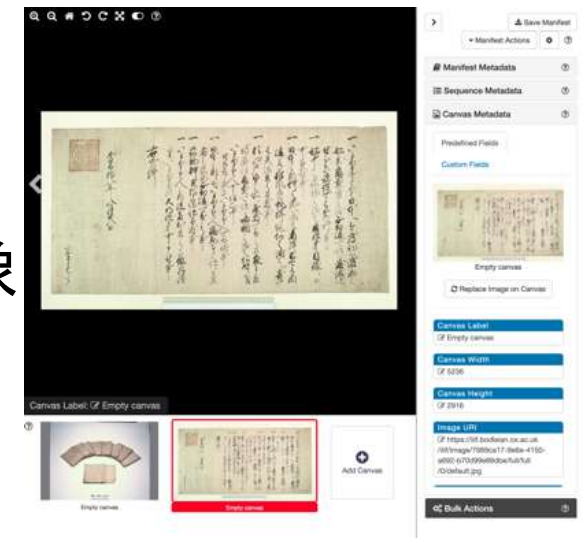
- IIIFへの理解を深めるためにマニフェストファイルの構造について説明したが、テキストエディタでマニフェストファイルを作成している機関はほとんどないと思われる
- コンテンツ管理システムのモジュール(後述)や、下記のようなツールを使用して作成することになる

- IIIF Manifest Editor

- <http://iiif.bodleian.ox.ac.uk/manifest-editor>
- UIで資料のメタデータや、資料の各頁の画像を選択してマニフェストファイルを作成

- Metadata Transformation Tools

- <https://github.com/bodleian/dmt-metadata-transformation>



マニフェストファイルの作成方法 [2/2]

- Presentation API Validator
 - <https://iiif.io/api/presentation/validator/service/>
 - IIIF公式のツールで、マニフェストファイルがPresentation APIの決まりに沿っているかチェックしてくれる

III F対応 デジタルアーカイブの 実装事例

実装事例 [1/2]

一般的な構成:

コンテンツ管理システム(CMS) (+ Image API対応画像サーバ)

- CMSにIIIF対応のためのモジュールをインストールすることで、IIIFでの資料の扱いが可能となる
- Image API対応画像サーバについては、26頁参照
- 画像サーバの役割を果たすCMSのモジュールも存在するが、高精細画像の配信には画像サーバの利用が望ましい

• Omeka S (CMS) を利用した事例

- 中村覚. Omeka Sを用いたIIIF対応デジタルアーカイブ構築の実際. オンラインワークショップCMSを利用したデジタルアーカイブの構築, 2020. <http://hdl.handle.net/2241/00161416>

実装事例 [2/2]

- Drupalを用いたIIIF対応デジタルアーカイブ構築の試み
 - 和氣愛仁. Drupalを用いたIIIF対応デジタルアーカイブ構築の試み. オンラインワークショップCMSを利用したデジタルアーカイブの構築, 2020. <http://hdl.handle.net/2241/00161415>

- リポジトリ

- DSpace
- WEKO3 ?

デジタルアーカイブ機能

- 簡易的なIIIF Image server機能を搭載
 - 画像リソースをImage API経由で公開可能
- IIIF対応ビューワを搭載（今年度）
 - 搭載ビューワを検討：Universal Viewer、Mirador、IIIF Curation Viewer
 - manifest.jsonのプレビュー
 - 外部イメージサーバ利用機関を想定
- manifest.jsonの生成（今年度）

International Image Interoperability Framework (IIIF)
画像データ相互運用のための技術標準。
国内外の多くのデジタルアーカイブが採用。

Image API例：
[http://\(host\)/api/iiif/v2/\(resource\)/full/750,0/default.png](http://(host)/api/iiif/v2/(resource)/full/750,0/default.png)

Manifest.jsonのエクスポートボタンの表示

Manifest.jsonの場合はビューワアイコンを表示。
クリック時に別ウィンドウを開く

機関の貴重書資料の公開を強化

16

出典：国立情報学研究所. 次期JAIRO Cloud Update.図書館総合展, 2019.
https://www.nii.ac.jp/event/upload/libfair2019_booth.pdf

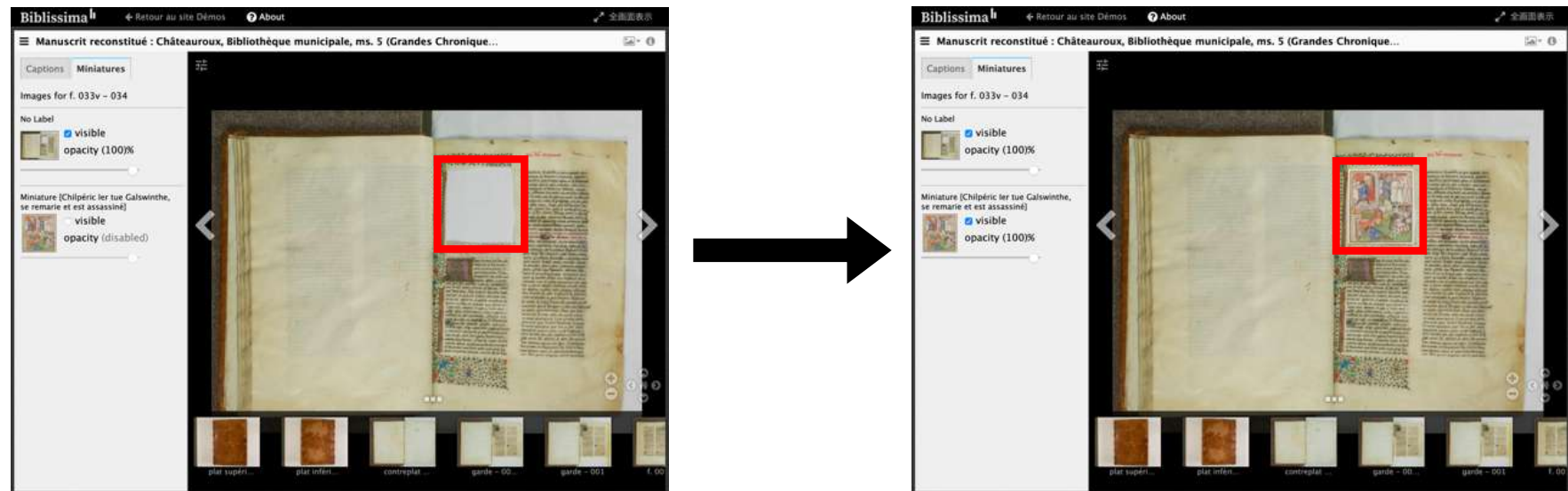
IIIFならではの 高度な資料活用

資料の再統合

Châteauroux の手稿（14世紀）の再統合

<https://demos.biblissima.fr/chateauroux/demo/>

- 一つのキャンバスに複数の画像を貼り付けることで、散逸されて所蔵されている資料の再統合を実現している
- レイヤ機能（キャンバスへの複数画像の貼り付け・表示）に対応しているビューワとして、Miradorが挙げられる

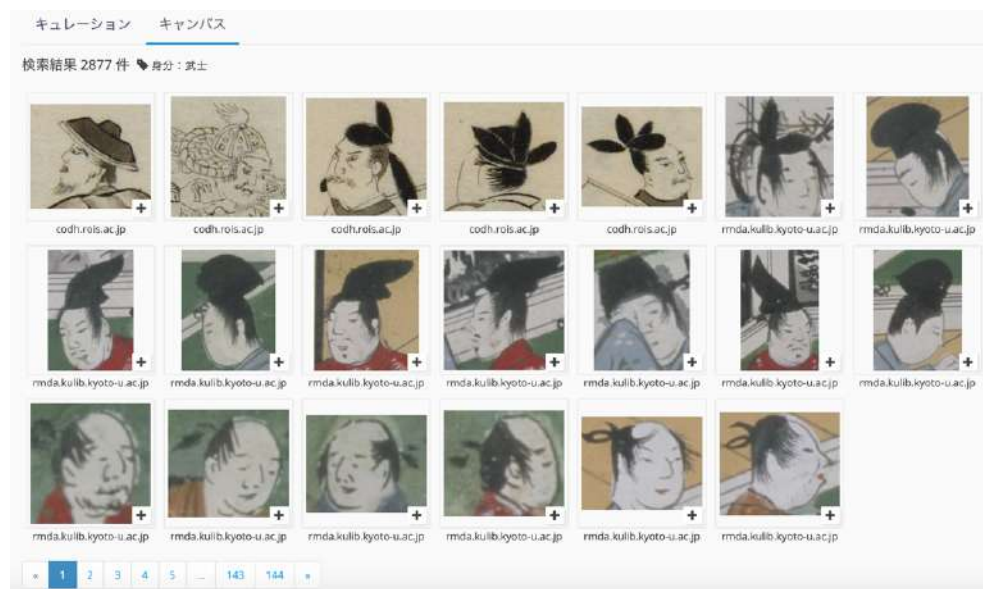


参考: バラバラになった各地の西洋中世写本断片をまとめて表示: IIIIFの活用例

<http://digitalnagasaki.hatenablog.com/entry/2017/07/03/030708>

IIIF Curation Viewer

- キュレーションという独自の構造をIIIFの世界に導入し、IIIF Image APIの仕組みを活用した部分画像へのリンクを収集を実現
 - テーマ別の画像コレクションを利用者が独自に作成し公開
 - 例:「顔貌コレクション」(日本美術の絵本・絵巻から顔の部分のみを収集)



出典: http://codh.rois.ac.jp/face/iiif-curation-finder/?select=canvas&where_metadata_a_label=%E8%BA%AB%E5%88%86&where_metadata_value=%E6%AD%A6%E5%A3%AB&lang=ja

参考: 北本朝展. オープンな画像の利活用を開拓するIIIF Curation Platform. カレンとアウェアネスE, E2301, 2020.
<https://current.ndl.go.jp/e2301>

III Fに関連するサイト・ ソフトウェア・ツールの 紹介

IIIF導入機関

IIIF公式サイトで導入機関のリストが提供されている

- Participating Institutions

<http://iiif.io/community/#participating-institutions>

- 完全なリストではない
- リンク先の多くは、デジタルアーカイブではなく、機関のトップページである

ソフトウェア、ツールの一覧(公式)

- **Awesome International Image Interoperability Framework (IIIF) (通称: awesome-iiif)**
 - <https://github.com/IIIF/awesome-iiif>
 - (ほとんど英語であるが、)IIIFに関連するソフトウェア・ツール・教材の一覧
 - IIIFについて学習したり開発するための教材
 - Tutorials, <https://github.com/IIIF/awesome-iiif#tutorials>
 - Videos and Slide Decks, <https://github.com/IIIF/awesome-iiif#videos-and-slide-decks>

ビューワ

- Universal Viewer

- <https://universalviewer.io/>



- Mirador

- <http://projectmirador.org/>



- Curation Viewer

- <http://codh.rois.ac.jp/software/iiif-curation-viewer/>

- 人文学オープンデータ共同利用センター(CODH)により開発

- 複数の資料から任意のページ・領域を抽出し、任意の順序で閲覧する機能

- その他については下記のリストを参照

- <https://github.com/IIIF/awesome-iiif#image-viewers>

文献

• ブログ・ウェブサイト

- 永崎研宣氏のブログ, <http://digitalnagasaki.hatenablog.com/>
- 神崎正英氏のブログ, <http://www.infocom.co.jp/das/loddiary/iiif/>
- 神崎正英氏のサイト, <https://www.kanzaki.com/works/2016/pub/image-annotator>

• 文献

- Snyderman, S., Sanderson, R., & Cramer, T. (2015). The International Image Interoperability Framework (IIIF): A community & technology approach for web-based images. In *Archiving Conference* (Vol. 2015, pp. 16-21). Society for Imaging Science and Technology.
- 永崎研宣. (2017). デジタル文化資料の国際化に向けて: IIIF と TEI. *情報の科学と技術*, 67(2), 61-66.
- 神崎正英. (2017). リンクの機能を柔軟に生かすデータのウェブ. *情報の科学と技術*, 67(12), 622-627.
- 永崎研宣. (2018). [C14] 国際的な画像共有の枠組み IIIF の課題と展望. *デジタルアーカイブ学会誌*, 2(2), 111-114.

まとめ

IIIF (International Image Interoperability Framework) : APIによって画像とそれに付随する情報を相互利用可能な形式で発信

- **IIIF Image API**: 画像の配信方法を規定
- **IIIF Presentation API**: 画像を表示するために必要な情報を規定
 - データモデルはWeb Annotation Data Modelに準拠している
 - 空白のキャンバス(資料の頁に相当)に画像やテキストをアノテーションとして付与する

APIとそのAPIに対応している数多くのソフトウェア、ツールによって非効率的な学術フロー、開発の重複を解決し、資料の利活用を促進している

終

補足: APIとは？

API (Application Programming Interface)

APIとは、あるコンピュータプログラム(ソフトウェア)の機能や管理するデータなどを、外部の他のプログラムから呼び出して利用するための手順やデータ形式などを定めた規約のこと。

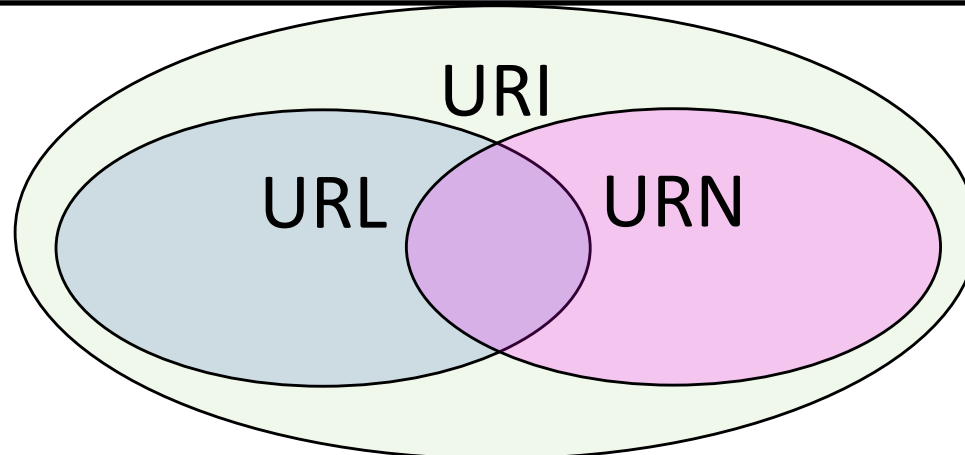
IT用語辞典 e-Words, <http://e-words.jp/w/API.html>

Image APIの場合

- コンピュータプログラム(ソフトウェア) → 画像サーバ
- 機能 → 画像への操作(切り取り、回転、色調)
- 管理するデータ → 画像
- 利用するための手順やデータ形式 → 画像をリクエストするURI構文

補足: URIとは？

- URL (Uniform Resource Locator)
 - リソースの場所(住所)を識別する
- URN (Uniform Resource Name)
 - リソースの名前を識別する
 - Web上からリソースが消失しても、一意で永続的な識別を行う
- URI (Uniform Resource Identifier) (統一資源識別子)



- URIが示すリソースは、機械で扱うデータ(HTML文書等)に限らず、エンティティ(人、組織、地名等)を示すことができる

補足: JSON

- JSON (JavaScript Object Notation)
 - 軽量のデータフォーマット
 - JavaScriptでオブジェクト(何かしらのデータ)を記述する書式
- JSONで表現するデータ型
 - 数値
 - 文字列(ダブルクォーテーションで括られる)
 - 真偽値(true、false(必ず小文字))
 - null
 - 配列(データのシーケンス)
 - オブジェクト

補足: JSONの配列とオブジェクト

- JSONの配列: 0個以上の値をカンマで区切って、[]で括る

```
["red", "pink", "blue", "orange", "yellow"]
```

- JSONのオブジェクト: コロンで区切られたキーと値のペアを、カンマで区切って0個以上列挙し、全体を{}で括る

```
{"name": "Nobita Nobi", "age": 10, "address": "Tokyo", "isPerson": true}
```

✗

```
{name: "Nobita Nobi", age: 10, address: "Tokyo", isPerson: true, }
```

- キーは必ず文字列で、ダブルクォーテーションで括られる
- 配列・オブジェクトにおけるカンマは、値あるいはペアを区切るためであり、最後尾の値あるいはペアのあとには不要である
- 括弧やカンマの前後のスペースの有無・改行はデータの読み込みに際して特に影響しない

補足: JSON例

```
{  
  "id": 27381222,  
  "name": "Kappa Ebisen",  
  "price": 121,  
  "tags": [ "Snack", "Prawn", "Popular" ],  
  "stock": {  
    "warehouse": 350,  
    "retail": 18  
  }  
}
```

補足: JSON-LD [1/2]

JSON-LD: Linked DataをJSONで記述するためのフォーマット

JSONのプロパティ名(キー)はローカルであり、データを広く共有するため、JSONにグローバルな名前(URI)を組み込む

従来のJSON

```
{ "name": "Mr. Bean",  
  "homepage": "http://mrbean.example",  
  "image": "http://mrbean.example/pic.png"  
}
```

JSON-LD

```
{ "http://schema.org/name": " Mr. Bean ",  
  "http://schema.org/url": { "@id": " http://mrbean.example"},  
  "http://schema.org/image": { "@id": " http://mrbean.example/pic.png" } }
```

補足: JSON-LD [2/2]

- コンテキストの導入: 簡略かつ人間にもわかりやすく

コンテキストの定義 <http://json-ld.org/contexts/person.jsonld>に保存

```
{ "@context":  
  { "name": "http://schema.org/name",  
    "image": {  
      "@id": "http://schema.org/image",  
      "@type": "@id"  
    },  
    "homepage": {  
      "@id": "http://schema.org/url",  
      "@type": "@id" }  
  }  
}
```

JSON-LD

```
{ "@context": "http://json-ld.org/contexts/person.jsonld",  
  "name": " Mr. Bean ",  
  "homepage": { "@id": " http://mrbean.example},  
  "image": { "@id": " http://mrbean.example/pic.png" } }
```