

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 (筑波大学人文社会系)

waki.toshihito.fn@u.tsukuba.ac.jp

目次

- はじめに
- 第1部：概説
 - 1-1. International Image Interoperability Framework (IIIF)
 - [コラム] IIIF 画像 API に依拠しない画像を使った IIIF マニフェスト
 - 1-2. Cantaloupe: High-performance dynamic image server in Java
 - [コラム] Cantaloupe をローカルで動かしてみる
 - 1-3. Drupal
- 第2部：Drupal と Cantaloupe を使ったデジタルアーカイブシステムの開発
 - 2-1. 開発方針
 - 2-2. Drupal 8 用 IIIF モジュールの開発
- 第3部：システム体験
 - 3-1. Cantaloupe への画像登録
 - 3-2. システムへの資料情報登録
- おわりに

[はじめに](#) ▶

はじめに

Who am I?

専門

- 日本語学（現代日本語文法論／コーパス言語学）
- デジタル・ヒューマニティーズ（人文情報学）

業務として

- ネットワーク・サーバ管理
- 各種大学業務システムの開発、仕様策定、調達

WDB: アノテーション付与型画像データベースプラットフォーム

- Drupal / Cantaloupe / Zoomify / Linux / Apache / MySQL / PHP
- 異なる種類の言語資料を、共通のデータ構造とユーザインターフェイスで扱えるようにする
- IIIF画像APIによる画像の公開
- IIIFプレゼンテーションAPIによるマニフェスト（資料情報）およびアノテーションリストの公開
- IIIF画像へのアクセス認可制御

構築中のサイト

- Hieratic Database Project: <https://wdb.jinsha.tsukuba.ac.jp/hdb/>
- 近代日本語文典集成: <https://wdb.jinsha.tsukuba.ac.jp/jgt/>
(繪入智慧の環 初編下詞の巻 » 5ウ-6オ: <https://wdb.jinsha.tsukuba.ac.jp/jgt/gallery/furukawa1870-vol1-2/7>)
- Sumero-Akkadian Cuneiform Tablets: <https://wdb.jinsha.tsukuba.ac.jp/xsux/>

第1部：概説

IIIF、Cantaloupe、Drupal について簡単にご紹介します。

- 1-1. International Image Interoperability Framework (IIIF)
- [コラム] IIIF 画像 API に依拠しない画像を使った IIIF マニフェスト
- 1-2. Cantaloupe: High-performance dynamic image server in Java
- [コラム] Cantaloupe をローカルで動かしてみる
- 1-3. Drupal

[◀ はじめに](#)

[上へ](#)

[1-1. International Image Interoperability Framework \(IIIF\) ▶](#)

1-1. International Image Interoperability Framework (IIIF)

What is IIIF?

画像へのアクセスを標準化し相互運用性を確保するための国際的なコミュニティ活動、およびそこで策定されている技術的な枠組み。

<https://iiif.io/>

IIIF Image API 2.1.1

<https://iiif.io/api/image/2.1/>

※最新は 3.0 [3.0での変更点]

IIIF Presentation API 2.1.1

<https://iiif.io/api/presentation/2.1/>

※最新は 3.0 [3.0での変更点]

IIIF Authentication API 1.0

<https://iiif.io/api/auth/1.0/>

IIIF Content Search API 1.0

<https://iiif.io/api/search/1.0/>

仕様の日本語訳

国立国会図書館の上綱秀治さんによる膨大な仕事。感謝！

<http://www.asahi-net.or.jp/~ax2s-kmtm/spec.html>

◀ 第1部：概説

上へ [コラム] IIIF 画像 API に依拠しない画像を使った IIIF マニフェスト ▶

[コラム] IIIF 画像 API に依拠しない画像を使った IIIF マニフェスト

早田清冷氏 (東京外国語大学 アジア・アフリカ言語文化研究所 情報資源利用研究センター 特任研究員)

『清文彙書』 (乾隆16年[1751])

<https://manjuisabuhabithe.aa-ken.jp/>

マニフェスト

<https://manjuisabuhabithe.aa-ken.jp/iiif/manifest/manifest.json>

```
"sequences": [
  {
    "@type": "sc:Sequence",
    "label": "Current Page Order",
    "canvases": [
      {
        "label": "巻1 (1)",
        "@id": "https://manjuisabuhabithe.aa-ken.jp/canvas/1",
        "@type": "sc:Canvas",
        "height": 4912,
        "width": 7360,
        "images": [
          {
            "@id": "https://manjuisabuhabithe.aa-ken.jp/annotation/1",
            "@type": "oa:Annotation",
            "motivation": "sc:painting",
            "resource": {
              "@id": "https://manjuisabuhabithe.aa-ken.jp/images/01002.jpg",
              "@type": "dctypes:Image",
              "format": "image/jpeg",
              "height": 4912,
              "width": 7360
            },
            "on": "https://manjuisabuhabithe.aa-ken.jp/canvas/1"
          }
        ],
        "thumbnail": {
          "@id": "https://manjuisabuhabithe.aa-ken.jp/images/01002t.jpg",
          "format": "image/jpeg",
          "height": 300,
          "width": 450
        }
      },
      ...
    ]
  },
  ...
]
```

1-2. Cantaloupe: High-performance dynamic image server in Java

What is Cantaloupe?

Javaで書かれたIIIF対応画像サーバ。イリノイ大学図書館によって開発されている。

<https://cantaloupe-project.github.io/>

特徴

- 動作環境の広範さと取り扱いの簡便さ
 - ウェブサーバ内蔵
 - タイル形式画像の事前作成不要：jpegがそのまま使える
 - 最低限必要な設定の簡便さ
- 設定可能な項目のきめ細かさ
 - サーバサイド・クライアントサイド両方での緻密なキャッシュ処理
 - 特にサーバサイドでのデリバティブキャッシュ（加工後画像のキャッシュ）は効果絶大
 - デリゲートスクリプト：一部のサーバ内部処理をユーザが拡張可能（Ruby言語使用）
 - 特に、アクセス元IPアドレス等の様々な条件や、あるいは外部認証プロバイダとの関係によって、画像へのアクセス制御を行えることは特筆すべきポイント

◀[コラム] IIIF 画像 API に依拠しない画像を使った IIIF マニフェスト

上へ

[コラム] Cantaloupe をローカルで動かしてみる ▶

[コラム] Cantaloupe をローカルで動かしてみる

以下は手元に Java 実行環境がある方向けです。java が動けば、Windows でも Mac でも（もちろん Linux でも）同じように実行できるはずです。

1. 画像配置ディレクトリを決めます。ここでは Mac を想定して仮に `/Users/me/Desktop/images/` とします。画像配置ディレクトリ内に、事前に用意した画像ファイルを置いてください。
2. Cantaloupe サーバのアーカイブを公式サイト <https://cantaloupe-project.github.io/> からダウンロードし、画像配置ディレクトリ内に展開します。展開したディレクトリ名をここでは `cantaloupe-4.1.6` とします。
3. `cantaloupe-4.1.6` 内にある設定ファイルのサンプル `cantaloupe.properties.sample` を `cantaloupe.properties` という名前でコピーします。
4. `cantaloupe.properties` をテキストエディタで開き、139行目の `FilesystemSource.BasicLookupStrategy.path_prefix` に画像配置ディレクトリの名前をセットします。末尾のスラッシュが必要であることを注意してください。

```
FilesystemSource.BasicLookupStrategy.path_prefix = /Users/me/Desktop/images/
```

5. コマンドライン（ターミナル/コマンドプロンプト）から以下のようにして Cantaloupe を実行します。cd コマンドで移動する先のディレクトリ名は各自読み替えてください。

```
$ cd /Users/me/Desktop/images/cantaloupe-4.1.6
$ java -Dcantaloupe.config=cantaloupe.properties -Xmx2g -jar cantaloupe-4.1.6.war
```

6. ブラウザで以下のURLにアクセスしてください（`iiif/2/` は IIIF 画像 API バージョン2の「エンドポイント」）。画像ファイル名 `1.jpg` の部分は適宜読み替えてください。ファイル名の大文字と小文字は区別されます。無事画像が表示されればOKです。

```
http://localhost:8182/iiif/2/1.jpg/full/full/0/default.jpg
```

注意：jpegファイルの内部形式の違いによって、時々 500 Internal Server Error が発生するようです。エラーメッセージが "Not a JPEG file: starts with 0xff 0xff" 等と表示されます。この場合は、いったん当該ファイルをなんらかの画像ソフトで開き、別名で保存してから再度アップロードしてみてください。少なくとも和氣の環境では PhotoShop で保存し直したらうまくいきました。この問題のサーバ側での対処法は現在調査中です。

7. 画像情報は以下のURLで取得できます。

```
http://localhost:8182/iiif/2/1.jpg/info.json
```

8. サーバを止めるときは Ctrl+C を押してください。

1-3. Drupal

What is Drupal?

オープンソースの汎用CMS。極めて拡張性に富んだモジュールシステムにより、様々な種類のウェブサイトを構築できる。

<https://www.drupal.org/>

採用例

- ロンドン公式サイト: <https://www.london.gov.uk/>
- NASA: <https://www.nasa.gov/>
- オックスフォード大学: <https://www.ox.ac.uk/>
- 以前のホワイトハウス（現在は WordPress である模様）
- ...

Why Drupal?

メリット

- 汎用CMSとしての柔軟性：拡張モジュールを組み合わせることで非常に細かな作り込みが可能
- ユーザーアカウント管理：ユーザー役割と権限設定を厳密にすることにより緻密なシステム運用が可能
- デザインテーマ：基幹サイト等とデザインを共通化することによりブランディング等に貢献可能
- コミュニティの大きさ：特に英語圏では情報が非常に充実している

デメリット

- システム構築の手数の多さ：システムの引き継ぎが困難になる可能性
- 日本語情報の少なさ：日本での採用例の少なさ（本当に少ない？）

第2部：Drupal と Cantaloupe を使ったデジタルアーカイブシステムの開発

今回構築した、Drupal によるデジタルアーカイブ管理システムについてご紹介します。

- 2-1. 開発方針
- 2-2. Drupal 8 用 IIIF モジュールの開発

◀ 1-3. Drupal

上へ

2-1. 開発方針 ▶

2-1. 開発方針

開発意図

- Omeka S 相当の機能を持ったウェブアプリケーションを Drupal を使って開発する

コンテンツタイプ

- アイテム
- メディア

機能開発

- すべて Drupal の流儀に従う
- 対応するメディアはひとまず IIIF 画像（Image API 経由）のみ
- 別途 IIIF 画像サーバとして Cantaloupe を利用する
- JSON-LD データのエクスポート機能の実装：RESTful Web Services モジュール
- データの一括取り込み機能の実装：CSV Importer モジュール
- 既存の拡張モジュールで対応できない部分＝IIIF マニフェストの生成機能を独自モジュールとして開発

2-2. Drupal 8 用 IIIF モジュールの開発

独自モジュールとして **IIIF Node Type** と **IIIF Manifest** のふたつを開発。以下は IIIF Manifest により自動的に生成されるマニフェストの例。

<https://iiif.jinsha.tsukuba.ac.jp/ws/item/furukawa1870-vol1-1/manifest>

IIIF Node Type

- ふたつのコンテンツタイプを定義
 - **iiifnodetype_item**
 - Omeka における「アイテム」相当
 - 定義済みフィールド: Identifier / Attribution / Description / License / Logo ほか
 - **iiifnodetype_media**
 - Omeka における「メディア」相当
 - 定義済みフィールド: Identifier / Display title / IIIF image URI ほか
- いずれのコンテンツタイプも通常のDrupalの方式に従ってフィールドを拡張可能
- アイテムページにメディア表示順変更ブロックを提供。D&Dで並べ替え（マニフェストに即反映）
- アイテム一覧、メディア一覧をページ表示する Views ビューを提供

IIIF Manifest

- IIIF Node Type のサブモジュールとして実装（IIIF Node Type に依存）
- IIIF マニフェストの自動生成
- Mirador（画像ビューア）表示ブロックを提供
- マニフェストの各プロパティに結びつけるフィールドを設定可能
- マニフェストのメタデータプロパティに含めるフィールドを選択可能
- 他のコンテンツタイプを独自に定義し、それをマニフェスト生成に使用することも可能

依存モジュール

- REST UI: <https://www.drupal.org/project/restui>
- Pathauto: <https://www.drupal.org/project/pathauto>
- DraggableViews: <https://www.drupal.org/project/draggableviews>
- Computed Field: https://www.drupal.org/project/computed_field

管理ページ（管理者のみアクセス可）

- iiifnodetype_item フィールド管理
- iiifnodetype_media フィールド管理
- IIIF Manifest モジュール設定

第3部：システム体験

今回構築したシステムへのデータ登録作業を、参加者のみなさまに実際に体験していただきます。

- 3-1. Cantaloupe への画像登録
- 3-2. システムへの資料情報登録

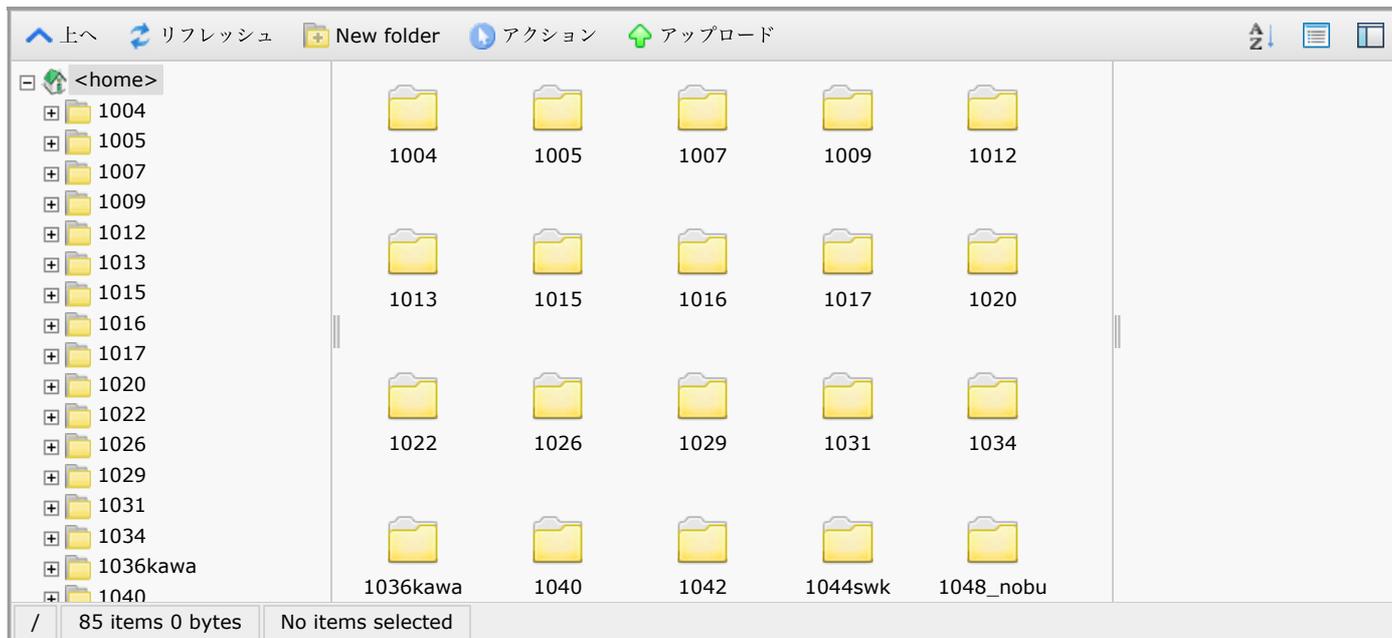
◀ 2-2. Drupal 8 用 IIIIF モジュールの開発

[上へ](#)

[3-1. Cantaloupe への画像登録](#) ▶

3-1. Cantaloupe への画像登録

Cantaloupe に画像を登録して、IIIF 画像 API を使って画像を取得してみます。システム上では Cantaloupe 画像サーバが稼働しており、特定のフォルダにファイルをアップロードするだけで、特定の URL を通じて画像を取得することができます。



1. 上のファイルマネージャー画面で、最上階層（<home> の中）に自分用のサブディレクトリを作ります。すでに `sample` というディレクトリがありますが、これと同じ並びに空のディレクトリを作成してください。ディレクトリ名は、他の方と衝突しないものにしてください。以下ではサブディレクトリ名として `sample` を例に説明します。
2. 自分で作ったサブディレクトリをダブルクリックして開き、ローカルに置いてある画像ファイル（jpg）をファイルマネージャー上にドラッグ&ドロップして画像をアップロードします。ここではアップロードしたファイルの名前を仮に `1.jpg` とします。
3. ブラウザで以下のURLにアクセスします。 `sample` はご自分のディレクトリ名に、 `1.jpg` はご自分のファイル名に置き換えてください。ファイル名の大文字と小文字は区別されます。無事に画像が表示されればOKです。なお `%2F` はスラッシュをパーセントエンコードしたものです。

```
https://iiif.jinsha.tsukuba.ac.jp/c/iiif/2/ws%2Fsample%2F1.jpg/full/full/0/default.jpg
```

IIIF画像APIにおける画像リクエストURI構文は以下のようになっています。

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

上の例で言うと、`{identifier}` が `ws%2Fsample%2F1.jpg` に、`{region}` が最初の `full` に、`{size}` が2番目の `full` に、`{rotation}` が `0` に、`{quality}.{format}` が `default.jpg` に、それぞれ該当します。

※注意：jpegファイルの内部形式の違いによって、時々 500 Internal Server Error が発生するようです。エラーメッセージが "Not a JPEG file: starts with 0xff 0xff" 等と表示されます。この場合は、いったん当該ファイルをなんらかの画像ソフトで開き、別名で保存してから再度アップロードしてみてください。少なくとも和氣の環境では PhotoShop で保存し直したらうまくいきました。この問題のサーバ側での対処法は現在調査中です。

4. また、以下のURLで、画像情報が表示されます。これは Cantaloupe が自動的に生成しています。

```
https://iiif.jinsha.tsukuba.ac.jp/c/iiif/2/ws%2Fsample%2F1.jpg/info.json
```

5. 画像URL末尾のパラメータ部分 /full/full/0/default.jpg をいろいろ変えてみます。以下は sample に入っている画像を、いくつかパラメータを変えて取得してみたものです。ワークショップ当日に口頭で若干説明を加えますが、詳しくは IIIF 画像 API 仕様 (4. Image Request Parameters / 日本語訳) を参照してください。

```
https://iiif.jinsha.tsukuba.ac.jp/c/iiif/2/ws%2Fsample%2F3.jpg/full/640,/0/default.jpg
```

```
https://iiif.jinsha.tsukuba.ac.jp/c/iiif/2/ws%2Fsample%2F2.jpg/full/!360,360/15/default.jpg
```

```
https://iiif.jinsha.tsukuba.ac.jp/c/iiif/2/ws%2Fsample%2F4.jpg/2000,500,1300,900/640,/0/gray.jpg
```

3-2. システムへの資料情報登録

Cantaloupe に登録した画像をまとめて、本システム上で資料情報として登録・公開してみます。

1. まず先に、必要なメディア（画像）を本システムに登録します。メディア一覧ページのタイトル直下に、「メディアを追加...」のリンクが出ていますので、これをクリックします。すると「iiifnodetype_mediaの作成」画面が出ます。別タブで開いた方が便利だと思います。
2. 必要な情報を入力します。

- **タイトル**：本システム上でのメディアのタイトル
- **Display title**：Miradorビューア上での画像のタイトル
- **Identifier**：画像の識別子（任意）
- **IIIF image URI**：IIIF画像のURI。先ほど Cantaloupe に登録したもの

なお、IIIF image URI は {scheme}://{server}/{prefix}/{identifier} の形式、すなわちファイル名までの文字列を入力してください。"/info.json" あるいは "/full/full/0/default.jpg" 等は入力しないでください。以下は例です。sample はご自分のディレクトリ名に、1.jpg はご自分のファイル名に置き換えてください。

```
https://iiif.jinsha.tsukuba.ac.jp/c/iiif/2/ws%2Fsample%2F1.jpg
```

3. 最後に保存ボタンをクリックします。うまくいけばメディアページが表示され、画像のサムネイルが読み込まれるはずです。
4. 以上を必要な回数繰り返して画像をすべて登録します。
5. 次に、アイテム（資料）の情報を本システムに登録します。所蔵資料一覧ページのタイトル直下に、「アイテムを追加...」のリンクが出ていますので、これをクリックします。すると「iiifnodetype_itemの作成」画面が出ます。別タブで開いた方が便利だと思います。
6. 必要な情報を入力します。以下は一部です。
 - **タイトル**：本システム上でのアイテムのタイトル（必須）
 - **Identifier**：アイテムの識別子（必須）
 - **Viewing direction**：Mirador ビューア上でのページ方向

なお、Identifier は本システム上のすべてのアイテムの中でユニークである必要があります。

7. Media の欄で、当該アイテムの画像として含めたいメディアにチェックを入れます。適宜スクロールして必要なものを選んでください。並び順はここでは気にしなくてOKです。
8. 最後に保存ボタンをクリックします。うまくいけば資料情報ページが表示されて、Miradorに画像が読み込まれるはずです。
9. メディアの並び順はここで調整可能です。必要に応じてドラッグ&ドロップで並べ替え、最後に下部の「順序の保存」ボタンをクリックしてください。
10. IIIF manifest URI をクリックするとマニフェストが表示されます。マニフェストURIは（Identifierがユニークであれば）固有の文字列になっています。このマニフェストを外部ビューア等に読み込ませることで外部ビューアで資料情報と画像を表示することが可能です。以下は Universal Viewer の公式サイト の URL です。ここにあるマニフェスト読み込みテキストボックスにご自分のマニフェストのURLを貼り付けて、マニフェストを読み込ませてみてください。

```
https://universalviewer.io/
```

なお、Universal Viewer で開くときには License の項目は必須のようです。何らかの文字列が入っていれば大丈夫です。

おわりに

汎用CMSのメリット

- 柔軟性：拡張モジュールを組み合わせることで非常に細かな作り込みが可能
 - 特定の目的に特化したカスタマイズ
- ユーザーアカウント管理：ユーザー役割と権限設定を厳密にすることにより緻密なシステム運用が可能
 - コンテンツへのアクセス制御等の細かなユーザー権限設計能力は大きな魅力
 - さらに：Cantaloupe のデリゲートスクリプトによる、Drupalのセッション情報を利用した画像へのアクセス認可制御
- デザインテーマの共用：基幹サイト等とデザインを共通化することによりブランディング等に貢献可能
 - これまで明確には意識されてこなかった？しかしウェブメディアとしては非常に重要

デメリット（と言いたくないが）の解消に向けて

- システム構築の手数の多さ：システムの引き継ぎが困難になる可能性
 - デジタルアーカイブ管理システム（仮称DAMS）の開発と、デジタルアーカイブ（DA）の構築を別レイヤーとして分ける
 - DAMSの開発：Omaka 相当のシステムを Drupal で開発。業者発注も可能
 - DAの構築：開発したDAMS を使って DA を構築、運用。組織内で運用可能なレベル。DAMS の改修は低レベルレイヤーに付託
- 日本語情報の少なさ：日本での採用例の少なさ（本当に少ない？）
 - 今日のような活動を通じて Drupal も Cantaloupe も採用例を増やし、日本語の情報を充実させていきたいです！

参考情報（1）

CSS Editor（Drupal 拡張モジュール）

https://www.drupal.org/project/css_editor

デザインテーマやモジュールによって提供される CSS を、ブラウザ上から直接編集できる。レタッチ的に一部のデザインを修正したいときに便利。テーマやモジュールのバージョンアップ時のパッチの再適用作業を気にしなくて済むのが大きなメリット。

参考情報（2）

Islandora: Open source digital asset management

Official site: <https://islandora.ca>

Documentation: <https://islandora.github.io/documentation/>

Sandbox: <http://future.islandora.ca>

Drupal, Cantaloupe, Fedora（リポジトリ管理）、Solr（全文検索）等々をパッケージ化したデジタル資産管理システム。全体の構成は以下を参照。

<https://islandora.github.io/documentation/technical-documentation/diagram/>

かなり大規模な構成ではあるが、構築済みの VirtualBox 仮想マシン（ゲスト OS は Ubuntu）も配布されているので、これを使えば比較的手軽に利用開始できる。

その他、デジタル資産管理システム Digital Asset Management というキーワードで検索すると色々ヒットするので、必要に応じてこちら方面を探してみるのもひとつの方法。

当日の参加者が体験作成したサンプルアーカイブ

所蔵資料一覧 | デジタルアーカイブ

iiif.jinsha.tsukuba.ac.jp/ws/item

デジタルアーカイブサンプルサイト 所蔵資料一覧 メディア一覧 ワークショップ目次 アカウント情報 ログアウト

ホーム / 所蔵資料一覧

所蔵資料一覧

アイテムを追加...

Sort by Created 順序 Desc Items per page 20 Apply

			
1182	1192 タイトル	1188_item	1185-test
			
1155-gazou2 鯉絵	2001 清少納言 5巻, 巻3	1180	1155-gazou1 女礼式歌合
			
ちゅーりっぶさん ちゅーりっぶさん喜怒哀楽	がまじゃんばー がまじゃんばー喜怒哀楽	1154_flower 花	繪入智慧の環 初編上 詞の巻 明治3(1870)年初版、古川正雄著。全3巻。
			