

# An Implementation Method of *GACA*: Global Art Collection Archive

Yosuke TSUCHIYA<sup>a,b1</sup> and Naoki ISHIBASHI<sup>a</sup>

<sup>a</sup> *Graduate school of Data Science, Musashino University*

<sup>b</sup> *IT System Group, Ishibashi Foundation*

## Abstract

In this paper, an implementation method of GACA, Global Art Collection Archive, is proposed. Each museum maintains their own archives of art collections. GACA dynamically integrate those collection data of artworks in each museum archive and provide them with REST API. GACA works as a integrated data platform for various kinds of viewing environment of artworks such as virtual reality, physical exhibitions, smartphone applications and so on. It allows users not only to view artworks, but also to experience the creativity of artworks through seeing, feeling, and knowing them, inspiring a new era of creation.

**Keywords.** museum database, open data, multidatabase system

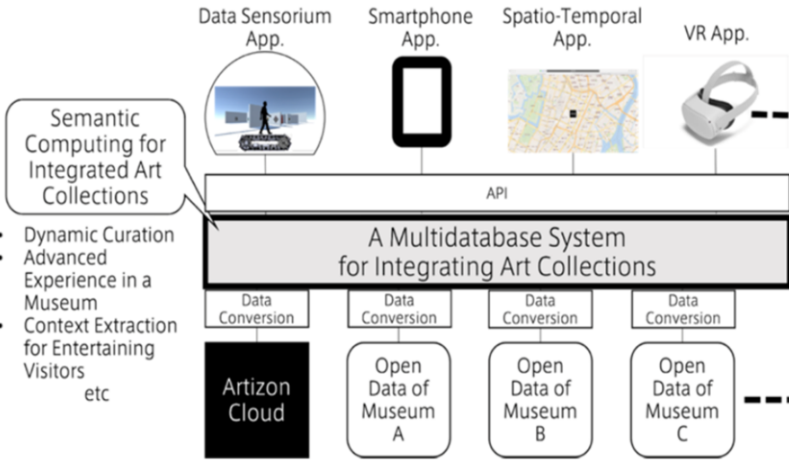
## 1. Introduction

In recent years, many museums put large efforts to establish digital archives of their art collections. Some museums such as the Metropolitan Museum of Art in New York[1], the Paris Musées[2], and the Louvre Museum[3], provide their archives on art collections as open data. These archives contain various types of information (such as information about the collections and exhibitions) in various media formats (such as text, images, video, audio, and 3D models). The archives of these museums consist of different media and different genres of artworks, and multidatabase system approach [4,5,6,7] that seems applicable to integrate such heterogeneous archives. *Artizon Cloud*[8] is a multidatabase system that integrates various archives of art collection data and enables them to use inside the museum and public areas while properly handling issues such as copyrights.

By integrating collection data, new types of art experience could be implemented in both physical and virtual spaces. Art Sensorium Project[9], as shown in Figure 1, was launched in Musashino University with focusing on two key technologies as follows: 1) a multidatabase system architecture to integrate multiple art collections, 2) virtual space design and implementation for the Data Sensorium[10]. In particular, personalized art exhibitions where artworks are selected from a museum archive and displayed based on the viewer's tastes and viewing tendencies could be implemented [11].

---

<sup>1</sup> Corresponding Author: Yosuke Tsuchiya, Graduate School of Musashino University, 3-3-3, Ariake, Kotoku, Tokyo, Japan: E-mail: g2251002@stu.musashino-u.ac.jp



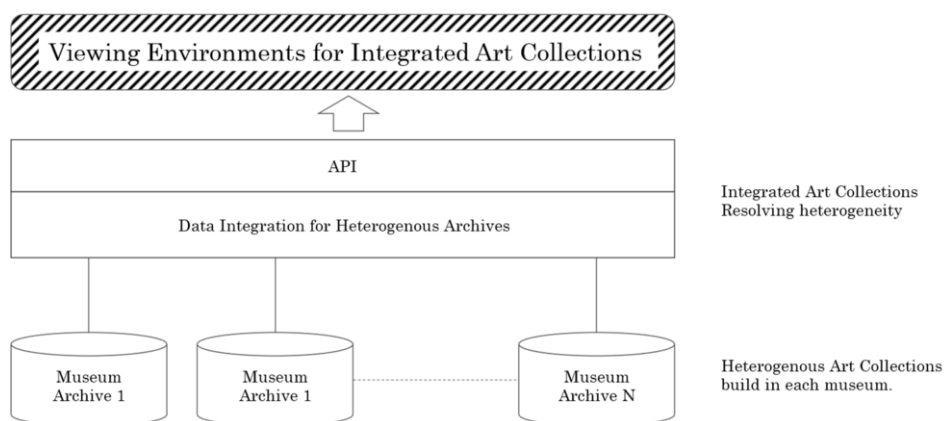
**Figure 1.** A System Architecture of Art Sensorium Project

The primary focus of the Art Sensorium Project is the design and implementation of a multidatabase system that integrates art collections. Therefore, in this paper, an implementation method of *GACA*, Global Art Collection Archive, is proposed. Each museum creates its own archives of art collections, then *GACA* integrates these archives and provides appropriate data for use in virtual reality, Data Sensorium, personal fabrication[12], and other applications. The structure of the paper is as follows: Section 2 reviews related researches and highlights the challenges that need to be addressed. Section 3 outlines the *GACA* architecture, designed to integrate with an arbitrary number of museum archives. Section 4 details the implementation of *GACA*, focusing on its integration with three specific museum archives. Section 5 presents two experiments conducted to verify the system's functionality and effectiveness.

## 2. Related Researches

There are several existing integrated archives of art collections, including Japan Search[13] and the Heritage Connector[14] in the UK. These archives receive government subsidies and contain art collections from museums within their respective countries. Google Arts & Culture [15] is another example of an integrated archive. It provides art collections and connects them to other web services provided by Google such as Geo Locations and Augmented Reality. These existing integrated archives of art collections have issues in terms of data heterogeneity and collection coverage.

The first issue is data heterogeneity, which can manifest in two ways: the heterogeneity of the data structure and the heterogeneity of the data notation. In terms of data structure, art collection data of each museum are typically organized according to each own rules. This results in missing data items (e.g., an item that exists in one museum but not in another) and non-uniform data types (e.g., a serial ID of a work may be an integer in one museum but a string in another).



**Figure 2.** Idea of GACA

The other is about the heterogeneity of data notation, such as the language notation of the artist's name, unit of artwork size, and so on. For example, the same "artist name" may be listed in multiple languages in one museum, while in another museum, only listed in the local language. Additionally, the size of a work may be listed in centimeters in one museum, but in inches in another.

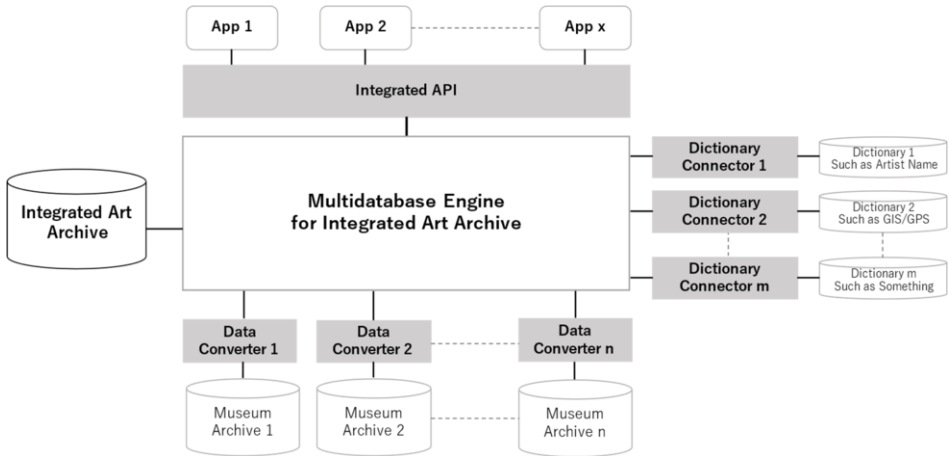
The second issue is collection coverage. Covering art collections from all over the world is a challenging task. To achieve this, an environment should be created where art collection data from each museum can be easily shared online. Japan Search and the Heritage Connector are federally subsidized integrated collection archives that provide collection data from museum archives via API linkage or CSV. However, their coverage is limited to data from domestic museums.

Google Arts & Culture is a cloud service provided by Google. Museums can choose to manually register their collections with Google Arts & Culture, which can then be linked to other Google services such as Google Maps and AR. However, not all museums provide all of their collection data to Google Arts & Culture (although some, such as the MET, do provide API integration). Many museums register their collections with Google Arts & Culture for public relations purposes, such as showcasing famous painters. This means that less famous, but still significant and important artworks are not accessible.

Our goal is to create a viewing environment that allows users not only to see famous artworks, but also to experience the creativity of artworks through seeing, feeling, and knowing them, inspiring a new era of creation. To achieve this, GACA is designed as a data platform that integrates various archives of art collection data maintained by each museum, making them accessible through a range of devices and applications (as shown in Figure 2). To facilitate seamless integration and use of these data within the devices and applications, GACA will be implemented as a multidatabase system that addresses the previously mentioned issues of heterogeneity and collection coverage.

### 3. GACA Architecture

GACA is designed as shown in Figure 3 to address the issues with existing integrated archives, such as data heterogeneity and collection coverage. Specifically, GACA



**Figure 3.** A System Architecture of GACA

consists of five main components, Multidatabase Engine for Integrated Art Archive, Integrated Art Archive, Data Converters, Dictionary Connectors and Integrated API.

### 3.1. Multidatabase Engine for Integrated Art Archive

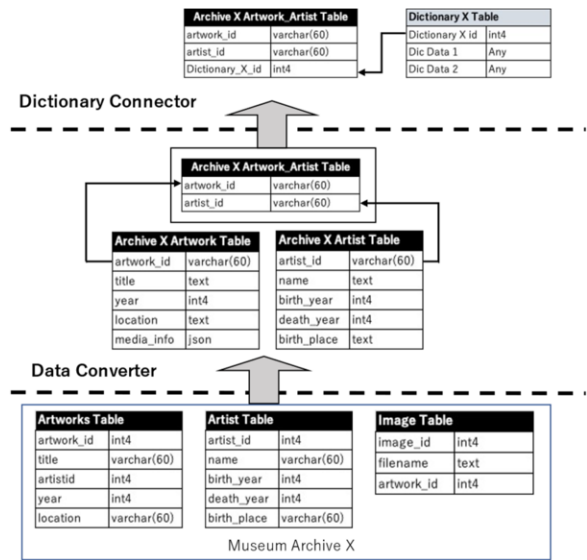
The Multidatabase Engine for Integrated Art Archive serves as the core of GACA. It incorporates various curatorial functions of data from the Integrated Art Archive, utilizing methods such as image recognition, machine learning, spatial and temporal operations and so on. Multidatabase Engine sends requests based on those methods to the Integrated Art Archive, and receive integrated art collection data of each museum as response from the archive.

### 3.2. Integrated Art Archive

The Integrated Art Archive serves as a central location where art collection data from various museum archives are stored. As described below, Data Converters are used to store the art collection data of each museum with converting to a common schema. Dictionary Connector adds a unique key to those art collection data to connect it to the dictionary. The Integrated Art Archive utilizes these unique keys to provide the integrated art collection data in response to requests from the Multidatabase Engine.

### 3.3. Data Converters

The Data Converters convert the data schema of the art collections in each museum archive into a common data schema. Each museum builds its own archive, so each museum has a different schema for its collection data. For example, in one museum archive, artwork and artist information are maintained in separate tables, and each assigned a unique ID. In contrast, another museum archive maintains artworks and artists on the same table but does not assign a unique ID to artists. Additionally, each museum retains different types of media data. For instance, regarding multimedia data in a



**Figure 4.** A Data Flow in GACA

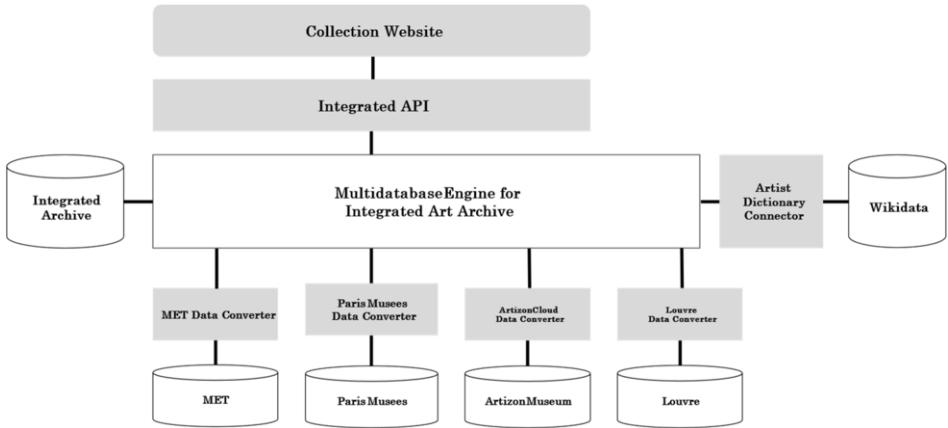
collection, one museum may only have images, while another may also manage audio and video in addition to images.

To address this heterogeneity among museum archives, GACA converts the collection data schema of each archive into a common collection data schema and stores it in the GACA Integrated Archive, as shown in Figure 4. The Data Converters generate three tables for managing collection data: a table of artworks, a table of artists, and a table of correspondence between artworks and artists. This allows the collection data from each archive to be combined at the meta-level, providing a comprehensive overview of the collections in the GACA Integrated Archive.

### 3.4. Dictionary Connectors

Dictionary Connectors obtain information on references (dictionary information) and assign unique information for data retrieval in GACA. Collection data contains a variety of information. For example, artist information includes the artist's name, place of birth, date of birth, and date of death. Artwork information includes the name of the work, year of creation, place of creation, materials, techniques, and dimensions. While this information is useful for searching collections, the notation and units differ within each museum's archives. For example, names may be written in the native language of the museum, and the locations and units of measurement for dimensions of works may vary (e.g., inches or centimeters). These differences make it difficult to search and compare information across different museums' archives.

To address the issue of heterogeneity of data notation, the Dictionary Connector connects to relevant dictionary databases, including artist notations, dimensions, time, and location data. The Dictionary Connector also generates a unique key for the art collection data from the dictionary database. It assigns this unique key to the collection data of each archive, connecting to the dictionary data. This enables cross-search and information extraction within the GACA.



**Figure 5.** An Implementation of GACA

By implementing Dictionary Connector and Data Converter, GACA enables to integrate of each museum archive dynamically, without selecting and limiting the collection data. The museum archives which store huge amounts of collection data could also easily be integrated into GACA by implementing the data converter. As a result, the GACA addresses the issue of collection coverage.

### 3.5. Integrated API

Integrated API works as the interface between Multidatabase Engine for Integrated Art Archive and various types of application such as virtual museum, Data Sensorium and so on. Integrated API designed as RESTful API with implementing a token authentication. When the Integrated API receives a request with a search key via GET or POST, API send the search key to Multidatabase Engine and receive the matching collection data. Integrated API return the collection data to the application with formatted in a supported data format such as JSON, CSV, or XML.

## 4. An Implementation of GACA

A GACA prototype system was implemented as shown in Figure 5. It is connected to four museum archives: the MET Collection, the Paris Musées Collection, the Artizon Cloud, and Louvre Collection. These archives are independently implemented and some of them offer open data with REST API. In addition, GACA is connected to an artist dictionary on Wikidata[16], which includes notation of artist names in five different languages: English, French, Chinese, Korean, and Japanese. As a result, GACA has integrated approximately over one million art collections from the four archives.

### 4.1. MET Data Converter

As shown in Figure 4, the role of data converters is to generate three tables with a common schema. The Metropolitan Museum of Art (MET) in New York City has approximately 480,000 artworks, documents, and other materials available as open data. This data can be accessed through REST API or CSV. However, the MET's collection

data is organized into a single table with both artist information and artwork information. Each artwork is given a local unique ID (Object ID), but the artists do not have IDs. On the other hand, each artist information includes a Wikidata URL, which has an Entity ID that uniquely identifies the artist information in Wikidata. To convert this data structure, the Data Converter for the MET Collection follows these steps:

1. Separate the data about the artwork and the artist from the collection CSV file.
2. Create a table of artworks using Object ID of the artworks as a key.
3. Extract Entity IDs from the Wikidata URLs of the artists.
4. Create a table of artist data using the Entity ID extracted in step 3 as a key.
5. Create a table of correspondence between the Entity ID of the artist and the Object ID of the artwork.

#### 4.2. Paris Musées Data Converter

Paris Musées is a public organization that oversees 14 museums in the city and has made the collections of approximately 360,000 items housed in these institutions available as open data. The collection data is shared among the 14 museums and can be accessed via a JSON-formatted API. In contrast to the MET API, which is based on RESTful, a GraphQL query must be generated to retrieve the collection data from Paris Musées. Each artwork and artist in the collection has a unique ID, and the artwork is also associated with the ID of the museum that owns it. To convert this data structure, the Data Converter for Paris Musées follows these steps:

1. Submit a GraphQL query to obtain information on the 14 museums in the Paris Musées network and retrieve the data in JSON format.
2. Using the ID of each museum as a key, submit a GraphQL query to retrieve the collection data stored in each museum and obtain the data in JSON format.
3. Separate the data related to artworks and artists from the data obtained in step 2.
4. Create an artwork table using the unique ID of each artwork as the key.
5. Create an artist table using the unique ID of each artist as the key.
6. Create a table of correspondences between the unique ID of each artwork and artist.

#### 4.3. Artizon Cloud Data Converter

The Bridgestone Museum of Art, which was founded in 1952, was reopened in 2020 as the Artizon Museum[17]. Artizon Cloud is a multidatabase system that contains various data archives related to artworks owned by the Artizon Museum. These archives include basic information of collection, evidential documents, multimedia (including images and sound), text, and event archives. Artizon Cloud controls the scope of collection offerings through three layers (Private Zone, Museum Zone, Public Zone) and rights relations. The Artizon Cloud Data Converter converts the data structure of the collection data published in the Public Zone of Artizon Cloud using the following steps:

1. Separate the data about artworks and the data about artists from the Artizon Cloud art collection.
2. Create a table of artworks using the artwork IDs as keys.
3. Create a table of artists using the artist's ID as a key.

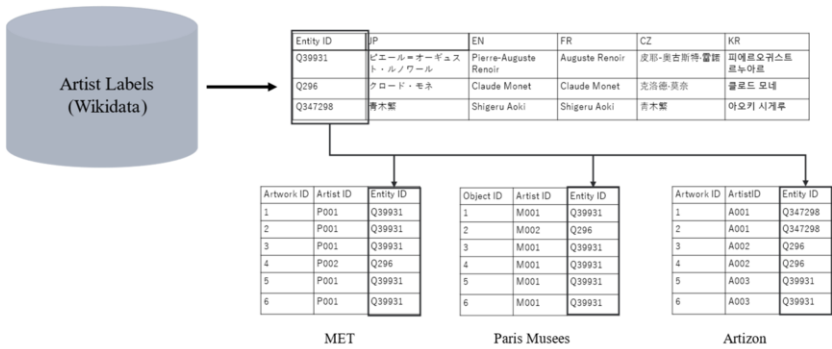


Figure 6. A Workflow of Dictionary Connector

4. Create a table of correspondences between the unique ID of each artwork and artist.

4.4. Louvre Converter

Louvre Museum is one of the largest museums in the world, and Louvre manages a large number of favored artworks. Louvre provides metadata of their artworks under the Etalab Open License[18], and they provide it in JSON format. Also, they provide the list of artworks by categories in CSV that includes their own identification numbers of each artwork. Therefore, the Louvre converter converts the data structure of the Louvre collection data using the following steps:

1. Download CSV files in each category of the artworks (Painting, Sculptures, Drawing, etc), and extract the Arkid (that uniquely identifies the artworks).
2. Then, download metadata in JSON format of each artwork using Arkid.
3. Create a table of artworks using the Arkid as a key.
4. Create a table of artists using the artist's ID as a key.
5. Create a table of correspondences between the artworks and the artists.

4.5. Artist Dictionary Connector

In this prototype system, Wikidata was utilized as the reference dictionary for artist notation. Wikidata is a collaborative, open data database that is compiled and normalized by volunteers. It is freely available to the public and has gained a reputation for credibility, receiving the Open Data Publisher Award in 2014. Additionally, its open-source nature has made it a central hub for datasets from various institutions, including libraries and museums.

Figure 6 demonstrates the process of creating the artist notation dictionary data from Wikidata. The MET has already used Wikidata URLs as references for artist information in its collection data, and the MET Data Converter employs Wikidata Entity ID as the artist ID of MET. Thus, the Dictionary Connector first uses this Entity ID to obtain the notation of the relevant artist in five languages and create a basic dictionary table. It then searches the basic dictionary data table for the artist notations of Paris Musées, Artizon Cloud and Louvre Museum, querying Wikidata for any artist notations



that do not match. These notations are obtained in the five languages and added to the basic dictionary table. Finally, the Dictionary Connector searches the respective Artwork-Artist tables generated by the Data Converter and assigns the corresponding artist a Wikidata Entity ID. This connects the artist information in each museum collection with the information in the notation dictionary, using the Wikidata Entity ID as the key.

#### *4.6. Integrated Art Archive*

Integrated Art Archive is implemented with a relational database management system (PostgreSQL version 12.3). As shown in Figure 3, art collection data, which is converted to a common data schema by Data Converter, are stored in artworks tables and artist tables. Also, Wikidata Entity ID of the artist's name is added to the artwork-artist table. Integrated Art Archive receives the query by SQL from Multidatabase Engine and returns the result set.

#### *4.7. Multidatabase Engine for Integrated Art Archive*

As a prototype of multidatabase engine for the Integrated Art Archive, a method to search art collections by the different languages of artist names was implemented. It works as following steps to retrieve the art collection data from Integrated Art Archive.

1. Receive all or a part of an artist's name from the Integrated API. The artist's name can be in any five languages (Japanese, English, French, Chinese, and Korean), obtained from the Dictionary Connector.
2. Search the received artist name in the Artist Dictionary to obtain the Wikidata Entity ID and notation of the artist name by five languages of the artist.
3. Submit a query to the Integrated Art Archive using the artist's Wikidata Entity ID obtained in step 2 as a key to retrieve the relevant artist's collection data. The query is submitted by SQL.
4. Receive the result set (integrated art collection data) from the Integrated Art Archive and return it to the Integrated API.

#### *4.8. Integrated API*

In this prototype, the Integrated API was implemented using Python 3.6.8 and the Flask framework for a web application. At this point, two endpoints have been implemented as follows: 1) an endpoint for artist name, which returns a set of Wikidata Entity ID and the artist's name in five languages from the Artist Dictionary, 2) an endpoint for Wikidata Entities ID, which returns a list of art collection data from the Multidatabase Engine for the Integrated Art Archive.

## **5. Experiments**

To confirm that the system is functioning as intended, two experiments were conducted: one to test the integrated search capabilities of the collections, and the other to test the operation of the Integrated API.

Artworks of Pierre-Auguste Renoir

Name: ピエール＝オーギュスト・ルノワール,Pierre-Auguste Renoir,Auguste Renoir,皮耶-奥古  
Birth: 1841 Death: 1919

API	Title	BeginYear	EndYear
LOUVRE	<a href="#">10 mai 1912. Les Collettes. Cagnes. à Théodore Duret</a>	1912	
LOUVRE	<a href="#">(14 janvier 1882). sans lieu. à X</a>	1882	
LOUVRE	<a href="#">Pièce autographe</a>		
met	<a href="#">Young Girl in a Blue Dress</a>	1890	1890
met	<a href="#">Two Young Girls at the Piano</a>	1892	1892
LOUVRE	<a href="#">La Danse à la campagne</a>		
(omitted)			
LOUVRE	<a href="#">Femme assise dans un intérieur entourée de personnages</a>		
ATZ	Nude		
Paris	<a href="#">Femme debout</a>	1906	
met	<a href="#">A Young Girl with Daisies</a>	1889	1889
ATZ	Terrace at Cagnes		
LOUVRE	<a href="#">Feuille d'études</a>		

Figure 7. A Result of Integrated Search by Artist name

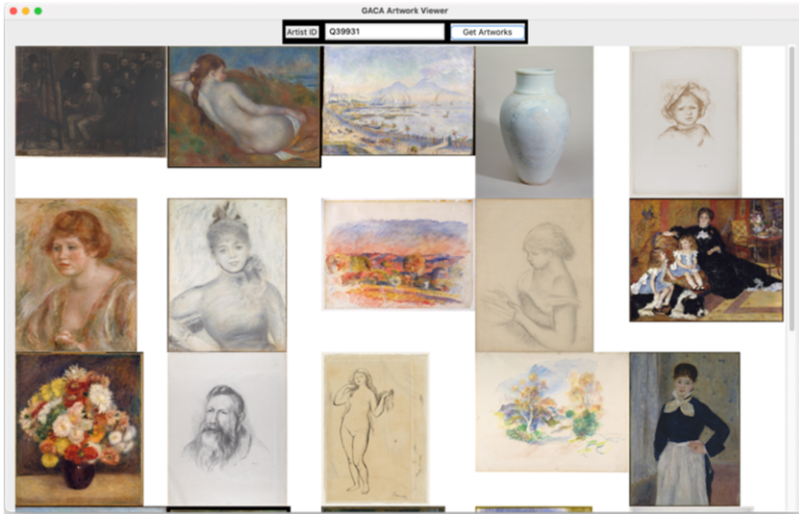
5.1. Experiment of Integrated Collection Search

An experiment was conducted to test the integrated search functionality of the system. During the experiment, it was confirmed that the following two points were working as expected: 1) searches for artists could be performed in five languages, and 2) the results showed that the works of the corresponding artists were retrieved from all four archives and displayed in a single list. Figure 7 shows some of the results, with works from the Metropolitan Museum of Art, Paris Musées, Artizon Cloud and Louvre displayed in the same list.

5.2. Experiments of Integrated API

As a test of the Integrated API, we created a simple artwork viewer in Python that displays images of artworks corresponding to the artist's search. This application displays images of the artist's artwork according to the following procedure:

1. The user send the Entity ID of the artist in the form at the top of the application.
2. The application accesses the API based on the Entity ID entered and obtains the JSON data of the list of works.
3. The application accesses the image URL of the artwork using JSON data.
4. The application download image of artworks and displays them.



**Figure 8.** Artwork Viewer from Integrated API

Figure 8 shows the artworks of Auguste Renoir displayed in the simple image viewer we created. Note that the API only includes images that are in the public domain, and only the corresponding artworks are displayed on the viewer<sup>2</sup>.

### 5.3. Observations and Future Issues

During the experiment, several issues became apparent. The first issue concerns the Integrated Collection Search, where some artworks—particularly those attributed to "anonymous"—could not be retrieved. Additionally, if the relevant artist's name was not available in a particular language on Wikidata, it would not appear in the search results. These problems could be resolved by integrating other dictionaries, such as those for geolocation, size units, or alternative artist databases. Further, more advanced methods for retrieving artworks, such as ontological modeling, should be explored.

The second issue relates to the application that uses the Integrated API. Given that GACA operates on a distributed system architecture, data retrieval can be unstable. This instability is exacerbated by the fact that GACA maintains URLs for images within each collection's data. These URLs must be accessed and downloaded each time. To correspond this issue, the Integrated Archive should store cached versions of artwork image files in various sizes (Small/Medium/Large).

## 6. Conclusion

In this paper, the system architecture for integrating heterogeneous art archives, Global Art Collection Archive (GACA), was proposed. GACA integrates approximately over one million artworks of four archives so far, and it provides accessibility via REST API. The most significant future effort is to create a viewing environment based on

<sup>2</sup> Images are cited from Met Collection: <https://www.metmuseum.org/art/the-collection>, and Paris Musées Collection: <https://www.parismuseescollections.paris.fr/en> (2022).

GACA that allows users to experience the creativity of artworks. We envision this environment not just as a place to view artworks, but as a space where users can fully engage with and be inspired by the artworks through seeing, feeling, and knowing them. We hope that this challenging effort will open a new era of creativity and innovation.

## Acknowledgment

Without collaborative research partnership between Musashino University and Ishibashi Foundation, this research would not even exist. We would like to express our utmost gratitude to Hiroshi Ishibashi, Taiji Nishijima, Kazunori Yamauchi, Takeshi Suzuki, Shoji Kometani and Tomohiro Kawasaki of Ishibashi Foundation for their great cooperation and support throughout the project period.

## References

- [1] The Metropolitan Museum of Art: The Met Collection, available via WWW, <https://www.metmuseum.org/art/collection> (2022)
- [2] Paris Musées: Les collections en ligne des musées de la Ville de Paris, available via WWW, <https://www.ParisMuséescollections.paris.fr/> (2022).
- [3] Musée du Louvre: Atlas database of exhibits, available via WWW, <http://cartelen.louvre.fr/> (2021).
- [4] Kitagawa, T. and Kiyoki, Y.: "The mathematical model of meaning and its application to multidatabase systems," Proc. 3rd IEEE Int. Workshop on Research Issues on Data Engineering: Interoperability in Multidatabase Systems, p.130–135 (1993).
- [5] Kiyoki, Y. and Kitagawa, T.: "A metadatabase system supporting interoperability in multidatabases", Information Modelling and Knowledge Bases, Vol.5, pp.287–298 (1993).
- [6] Kiyoki, Y., Kitagawa, T. and Hitomi, Y.: "A fundamental framework for realizing semantic interoperability in a multidatabase environment", Journal of Integrated Computer-Aided Engineering, Vol.2, No.1, pp.3–20 (1995).
- [7] Kiyoki, Y., Hosokawa, Y. and Ishibashi, N.: "A Metadatabase System Architecture for Integrating Heterogeneous Databases with Temporal and Spatial Operations," Advanced Database Research and Development Series Vol. 10, Advances in Multimedia and Databases for the New Century, A Swiss/Japanese Perspective, pp.158–165, World Scientific Publishing (1999).
- [8] Ishibashi, N.: "Artizon Cloud: A Multidatabase System Architecture for an Art Museum," Information Modelling and Knowledge Bases XXXIII, pp.323–331 (2022).
- [9] Ishibashi, N., Fukuda, T., Tsuchiya, Y., Enzaki, Y., Iwata H.: "Art Sensorium Project: A System Architecture of Unified Art Collections for Virtual Art Experiences," 33<sup>rd</sup> International Conference on Information Modelling and Knowledge Bases EJC 2023. (2023) (submitted)
- [10] Iwata, H., Sasaki, S., Ishibashi, N., Somlertlamvanich, V., Enzaki, Y., Kiyoki, Y.: "Data Sensorium: Spatial Immersive Displays for Atmospheric Sense of Place," Information Modelling and Knowledge Bases XXXIV, IOS Press, pp.247–257. (2022).
- [11] Fukuda, T., and Ishibashi, N.: "Virtual Art Exhibition System: An Implementation Method for Creating an Experiential Museum System in a Virtual space", Information Modelling and Knowledge Bases XXXIV, IOS Press, pp.38–47.(2022).
- [12] Gershenfeld, N., Gershenfeld, A. and Cutcher-Gershenfeld, J.: "Design Reality: How to Survive and Thrive in the Third Digital Revolution", Basic Book (2017).
- [13] Japan Search, available via WWW, <https://jpsearch.go.jp/> (2022).
- [14] Heritage Connector, available via WWW, <https://www.sciencemuseumgroup.org.uk/project/heritage-connector/> (2022)
- [15] Google Arts & Culture, available via WWW, <https://artsandculture.google.com/> (2022).
- [16] Wikidata, available via WWW, <https://www.wikidata.org/> (2022).
- [17] Artizon Museum, available via WWW, <https://www.artizon.museum> (2022).
- [18] Etalab Open License, available via WWW, <https://www.etalab.gouv.fr/wp-content/uploads/2018/11/open-licence.pdf>