REST Interoperability

Hugo Silva
Universidade do Minho
Gualtar, Braga, Portugal

The current work discusses interoperability, which refers to the ability of different systems, devices, or applications to communicate and exchange data seamlessly. Interoperability is essential in various industries, including finance, transportation, and healthcare. In healthcare, it helps to improve patient outcomes and reduce costs. REST (REpresentational State Transfer) is an architectural style for distributed hypermedia systems that has six guiding principles or constraints that must be satisfied for a service interface to be considered RESTful. These include uniform interface, client-server, stateless, cacheable, layered system, and code on demand. It is also highlighted some of the key trends and developments in interoperability, such as the adoption of common data standards, the use of APIs, and the implementation of interoperability frameworks.

Keywords: REST; Interoperability; Healthcare

1. Introduction

Interoperability refers to the ability of different systems, devices, or applications to communicate, exchange data and use that data to perform tasks seamlessly [1]. In other words, interoperability is the ability of various systems to work together effectively, and it is increasingly becoming critical in various industries.

For instance, in the financial industry, interoperability allows for seamless and secure transactions between different banks and payment platforms. In the transportation sector, interoperability enables different modes of transport, such as buses, trains, and airplanes, to communicate and exchange data for efficient transportation planning and management [2].

By ensuring that different systems and devices can work together, interoperability promotes innovation and competition, which ultimately leads to better products and services for consumers [3].

In the healthcare sector, interoperability refers to the ability of different healthcare systems, providers, and stakeholders to exchange patient health information (PHI) securely and efficiently. Standardized protocols, data formats and coding systems are essential for ensuring interoperability across different systems and devices.

These standards are also essential for ensuring data security and privacy, helping prevent unauthorized access, data breaches, and other security threats, which is crucial in healthcare, where the sensitive nature of patient health information requires strict data protection measures.

Interoperability enables healthcare providers to access and share patient data across different care settings and technology platforms, including electronic health records (EHRs), health information exchanges (HIEs), and mobile health apps.

The goal of interoperability in healthcare is to improve patient outcomes, reduce healthcare costs, and enhance the overall quality of care. By enabling healthcare providers to access complete and up-to-date patient information, interoperability helps to improve care coordination, prevent medical errors, and avoid unnecessary or duplicate tests and procedures [4].

Interoperability also enables patients to take a more active role in their healthcare by providing them with access to their health data, enabling them to share their data with providers, and allowing them to make informed decisions about their care. Overall, interoperability is an essential component of a modern, patient-centered healthcare system [4].

Some of the standards and architectures used in the healthcare sector to achieve interoperability are: HL7, FHIR, DICOM and REST.

2. Trends and developments

At present, some of the key trends and developments in interoperability include:

- Adoption of common data standards - To facilitate seamless data exchange between different systems and applications, there is an increasing emphasis on the adoption of common data standards. These standards ensure that data is structured and formatted consistently, making it easier to transfer and interpret [5].
- Use of APIs - Application Programming Interfaces (APIs) are becoming more prevalent as a means of facilitating interoperability between different systems. APIs allow different software applications to interact with each other, making it easier to share data and functionality [6].
- Implementation of interoperability frameworks - There are now a number of interoperability frameworks available that provide guidelines and best practices for achieving seamless interoperability between different systems. These frameworks typically address issues such as data security, data privacy, and data governance.

- Emphasis on open standards - The use of open standards is becoming increasingly important in achieving interoperability. Open standards are freely available and can be used by anyone, which makes it easier for different systems and applications to communicate with each other [7].
- Advancements in blockchain technology - Blockchain technology is being increasingly used to facilitate interoperability between different systems. Blockchain provides a decentralized platform for securely storing and sharing data, making it ideal for use in situations where multiple parties need to access and share data [8].

3. REST Framework

REST, short for REpresentational State Transfer, is an architectural style for distributed hypermedia systems first presented by Roy Fielding in his dissertation in the year 2000 [9].

REST has its own guiding principles that must be satisfied for a service interface to be considered RESTful.

3.1 Guiding principles

There are six guiding principles or constraints of the RESTful architecture [9]:

- Uniform Interface - by applying the principle of generality to the components interface, the system architecture can be simplified, and the visibility of the interactions can be improved. These four constraints can help obtaining a uniform REST interface:
  - Identification of resources - the interface must uniquely identify each resource involved in the interaction between the client and the server.
  - Manipulation of resources through representations – the resources should have uniform representations in the server response. API consumers should use these representations to modify the resources state in the server.
  - Self-descriptive messages – each resource representation should carry enough information to describe how to process the message. It should also provide information of the additional actions that the client can perform on the resource.
  - Hypermedia as the engine of application state – the client should have only the initial URI (Uniform Resource Identifier) of the application. The client application should dynamically drive all other resources and interactions with the use of hyperlinks.
- Client-Server - enforces the separation of concerns (separating the user interface concerns - client - from the data storage concerns - server)

which helps the client and server components evolve independently, improving scalability by simplifying server components.

- Stateless - mandates that each request must contain all the information necessary to understand and complete the request. This way, the server cannot take advantage of any previously stored context information on the server and the client application must entirely keep the session state.
- Cacheable - requires that a response should implicitly or explicitly label itself as cacheable (in which case the client application gets the right to reuse the response data later for equivalent requests and a specified period) or non-cacheable.
- Layered System - allows an architecture to be composed of hierarchical layers by constraining component behaviour (so that each component cannot see beyond the immediate layer they are interacting with).
- Code on Demand (Optional) - allows client functionality to extend by downloading and executing code in the form of applets or scripts. This simplifies clients by reducing the number of features required to be pre-implemented.

3.2 Resources

The key abstraction of information in REST is a resource, which can be any named information such as a document, an image, a temporal service, a collection of other resources or a non-virtual object.

The state of the resource, at any time, is known as the resource representation, which consists of [9]:

- the data
- the metadata, describing the data
- and the hypermedia links, that can help clients transition to the next desired state.

Resource Identifiers are used to identify each resource involved in the interactions between the client and server components, so, in this example: http://api.schemas.wrml.org/soccer/Player, the word ”Player” is a Resource Identifier [10].

Resource representations should be self-descriptive, and hypermedia of the resources are used to simultaneously represent their information and their controls (how they are to be processed). This can be achieved either explicitly (e.g., link and id attributes) or implicitly (e.g., derived from the media type definition and representation structure).

3.3 Resource Methods

Resource methods play a crucial role in REST as they facilitate the transition between two states of a resource.

They are commonly (and mistakenly) associated with HTTP methods (such as GET/PUT/POST/DELETE). However, Roy Fielding, the creator of REST, has not provided any recommendations on which methods should be used in each situation, instead, he only emphasizes that the interface should be uniform [9].

For example, if an application API uses the HTTP POST method for updating a resource, instead of the commonly recommended HTTP PUT method, it is still considered RESTful, as long as the interface is uniform.

Ideally, all the information necessary for transitioning the resource state should be included in the resource representation. This includes details on all supported methods and the format in which they will be presented.

4. Real-world use cases

There are many healthcare services and applications that use REST for interoperability and data exchange. Some of those examples are:

- Apple HealthKit - Apple HealthKit is a platform that allows users to track their health and fitness data from multiple sources, such as wearable devices and health apps. The HealthKit API uses REST to enable developers to securely access and share health data.

- Cerner Millennium - Cerner Millennium is a cloud-based EHR system used by healthcare organizations to manage patient data and clinical workflows. Cerner Millennium provides RESTful APIs that allow developers to create custom integrations and applications.

- Allscripts Sunrise - Allscripts Sunrise is another cloud-based EHR system used by healthcare organizations. Allscripts provides RESTful APIs that allow developers to create custom integrations and workflows.

- Practice Fusion - Practice Fusion is a web-based EHR system used by small and medium-sized healthcare practices. Practice Fusion provides RESTful APIs that allow developers to create custom integrations and applications.

- OpenMRS: OpenMRS is an open-source EHR system used in resource-constrained settings, such as developing countries. OpenMRS provides RESTful APIs that allow developers to create custom modules and applications.

5. Conclusion

Interoperability is a fundamental concept that enables different systems, devices, and applications to communicate and work together effectively.

Overall, the state of the art of interoperability is constantly evolving, with new technologies and best practices emerging all the time. The key to achieving seamless interoperability is to adopt common data standards, emphasizing on open ones, use APIs, implement interoperability frameworks, and take advantages of the latest technological advancements, such as blockchain technology.

In healthcare, interoperability plays a critical role in improving patient outcomes, reducing costs, and enhancing the quality of care. The use of standardized protocols and data formats is essential for achieving interoperability, and this trend is likely to continue as industries continue to adopt new technologies and systems.

RESTful APIs have multiple real-world applications in healthcare, such as facilitating the exchange of patient health information between different EHR systems, enabling real-time access to patient data for clinical decision-making, integrating healthcare applications with other third-party systems and services, and supporting research and innovation in healthcare. As healthcare continues to evolve and become more data-driven, RESTful APIs will play an increasingly important role in enabling providers to share data, improve patient outcomes, and deliver high-quality care.

References

[1] S. Lewis, *What is interoperability?*, TechTarget, Accessed on April 21, 2023. https://www.techtarget.com/searchapparchitecture/definition/interoperability

[2] *Interoperability White Paper*, United States Department of Transportation, Accessed on April 22, 2023. https://www.its.dot.gov/research_areas/WhitePaper_interoperability.htm

[3] G. Slover (January 12, 2023), *Interoperability is Important for Competition, Consumers, and the Economy*, Center for Democracy and Technology, Accessed on April 22, 2023. https://cdt.org/insights/interoperability-is-important-for-competition-consumers-the-economy/

[4] *Interoperability in healthcare*, IMB, Accessed on April 22, 2023. https://www.ibm.com/topics/interoperability-in-healthcare

[5] *Why Data Standards are Essential for True Interoperability and Ease of Integration*, The Clinician, Accessed on April 24, 2023. https://theclinician.com/blog/why-data-standards-essential-true-interoperability-ease-integration

[6] (April 20, 2023), *The benefits of APIs and interoperability*, The Clinician, Accessed on April 24, 2023. https://www.beckershospitalreview.com/the-benefits-of-apis-and-interoperability.html

[7] *Interoperability and Open Standards*, European Committee for Interoperable Systems, Accessed on April 24, 2023. http://www.ecis.eu/open-standards/

[8] (April 19, 2023), *What Is Blockchain Interoperability?*, Chainlink, Accessed on April 24, https://chain.link/education-hub/blockchain-interoperability

[9] L. Gupta (April 7, 2022), *What is REST*, REST API, Accessed on April 24, 2023. https://restfulapi.net

[10] M. Masse, *Chapter 4. Metadata Design*, O'Reilly Media, Accessed on April 24, 2023. https://www.oreilly.com/library/view/rest-api-design/9781449317904/ch04.html