A Model and Architecture for Building a Sustainable National Open Government Data (OGD) Portal

Lukman Lamid Idowu
National Information Technology Development Agency
28, PortHarcourt Crescent, Area 11, Garki, Abuja
Nigeria
lukmanlamid@yahoo.com

Isa Ibrahim Ali (PhD)
National Information Technology Development Agency
28, PortHarcourt Crescent, Area 11, Garki, Abuja
Nigeria.
isaaliibrahim@hotmail.com

Usman Gambo Abdullahi (PhD)
National Information Technology Development Agency
28, PortHarcourt Crescent, Area 11, Garki, Abuja
Nigeria.
drusmanga@gmail.com

ABSTRACT

Open Government Data (OGD), an extension of e-Government, so claimed, has become a major factor in the United Nations e-Government Development Index (EGDI) and to achieving the global sustainable development goals (SDGs). This is based on the argument that opening up government data has tremendous social, political and economic benefits. However, if data is open in a format that is not useful to data prospects, it is as good as data that is not open at all. Most governments around the world, especially in the developing countries, have concentrated their efforts around transparency in governance as a major reason for open data. They have only succeeded in opening and publishing data on government Web/open data Portals in unstructured formats. This has prevented innovative use and reuse of data by developers/entrepreneurs and the public for good governance, economic, environmental and social development. Also, access to quality and accurate data as well as how to address privacy issues are critical to achieving the promises of OGD. In addition, the institutional politics around data ownership among government agencies and their readiness to open actual data are critical to national OGD portal.

This paper develops an integrated Open Government Data model & architecture based on literature reviews of secondary data to address issues around data format, quality, accuracy and ownership. While addressing these challenges, the goal of the architecture is to guide government in implementing a sustainable OGD portal with linked data as the output being the highest level of open data format. The connection of related data makes data effectively reusable to create public value.

CCS CONCEPTS

• **Information Systems → Enterprise Information Systems**; *e-Government; Open Government* Data

KEYWORDS

Open Government Data (OGD), OGD Dimensions, Linked Data, OGD Model and Architecture, OGD Platform, OGD Portal

1 INTRODUCTION

Open Government Data is a new concept with data reusability as one of its major goals to ensure data is used and reused for creating innovative services that will improve quality of human lives. The term Open Government Data in this paper refers to data that makes the government as a whole more open (that is, more publicly accountable and transparent), and also politically neutral motives that encourage innovation and value creation through easy and free distribution, use and reuse of government data in line with the general guidelines and principles on OGD. The figure below symbolizes the two broad categorized reasons behind OGD.

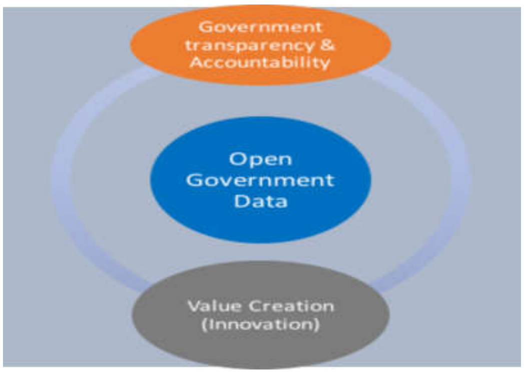

Figure 1: The two dimensions to open government data

This means the possibilities of open government data are vast; it creates opportunities not only to make the government transparent and accountable to its people but also support the creation of values and innovations by unlocking insights from data. Therefore, access to relevant, structured, accurate public information by the people and service applications has become increasingly important for public data reuse for value creation. For instance, the most relevant way to access data from a software developer's perspective is through an API. API enables access to open government data by interested parties; combine them (linked open data) and produce new information and mashup applications for a better society [1].

1.1 Objective: The overall objective of the paper is to develop all-inclusive model & architecture of OGD portal implementation as one of the tools for achieving SDGs. The specific objectives of this paper are:

- review of the technological trends that make open government data possible;
- develop a creative Open Government Data model and architecture for building a sustainable open government data portal for Nigeria and beyond based on related literature reviews;
- Establish a baseline framework that will initiate further research work in preparing toward accomplishing a widely accepted Open Government Data architecture based on standards

1.2 Delimitation: This paper is based on secondary data and only describes the conceptual model and architecture to implement a sustainable open government data. Open middleware for real-time unstructured and structured data from IoT solutions and social media contents which is part of the conceptual OGD model will not be described. Also, a detailed description of linked data concept is beyond the scope of this paper.

2 LITERATURE REVIEW

This section briefly describes current technological evolutions and achievements that meet OGD motivations. OGD is a new concept and these achievements were not initially targeted at OGD. They are however found to be perfect matches for achieving the motivations and goal of OGD. It describes different stages in the web evolution that made linked data achievable. In addition, it describes OData, one of the latest protocols for publishing and consuming open data using web standard technologies as well as OGD stage model.

2.1 Effort Towards Linked Data and Semantic Web

2.1.1 The world wide Web (WWW): The WWW is designed for publishing and accessing resources and services across the Internet. The Web is based on three main standard technological components: The HyperText Markup Language (HTML), a language for specifying the contents and layout of pages as they are displayed by web browsers; Uniform Resource Locators (URLs), also known as Uniform Resource Identifiers (URIs), which identify documents and other resources stored as part of the Web; A client-server system architecture, with standard rules for interaction (the HyperText Transfer Protocol – HTTP) by which browsers and other clients fetch documents and other resources from web servers [2].

2.1.2 Web Services: The HTML in the WWW is inadequate for programmatic interoperation and there was increasing need to exchange many types of structured data on the Web. This led to the creation of Extensible Markup Language (XML) for representing data in a standard, structured and application-specific formats. In principle, data expressed in XML is portable between applications since it is self-describing and any operation on a resource can be invoked using one of the GET or POST methods.
XML is not a complete solution for programmatic interoperation. JavaScript Object Notation (JSON) file format is becoming an alternative to XML because it is generally easier to read. JSON parsing is generally faster than XML parsing while XML is stricter and has support for schemas and Namespaces [3]. Therefore, the two formats are very important to software/application developers from open data application perspectives.

2.1.3 Linked Data & Semantic Web: The Web outstanding success is hinged on the openness of its system architecture, the ease with which content can be published and the suitability of its hypertext structure for organizing many types of information. However, its hypertext model is susceptible to dangling links, thereby causing frustration for users. The invention of search engines was an alternative for finding information on the web but it is not a perfect solution at producing what the user specifically intends. Another approach to this is Resource Description Framework (RDF) which was intended to produce standard vocabularies, syntax and semantics for expressing metadata and to encapsulate that metadata in the corresponding web resources for programmatic access. RDF makes it possible to represent data in a form that makes it easier to combine data from multiple sources. RDF can be stored in both XML and JSON. It encourages the use of URLs as identifiers, which provides a convenient way to directly interconnect existing open datasets on the Web to form linked data. RDF is still not widespread, but it has been a trend among Open Government initiatives, including the British and Spanish Government Linked Open Data projects. The inventor of the Web, Tim Berners-Lee, has recently proposed a five-star scheme that includes linked RDF data as a goal to be sought for open data initiative [4]. Rather than searching for words that occur in web pages, programs can then, in principle, perform searches against the metadata to compile lists of related links based on semantic matching. Collectively, the web of linked metadata resources is what is meant by the semantic web [2].

2.2 APIs for Consuming Open Data

Once linked open data is made available, there is need for it to be consumed by different applications in a programmatic way. APIs provide interface specifications that allow for machine to machine querying (i.e. application to application), essentially removing the barriers to access data. This ensures open data is innovatively provided as a service for economic benefits. The popular and highly suitable approach to delivering web APIs for open data initiative is Representational State Transfer (REST). REST with JSON has become the favorite of developers and API owners, because it is easier to both deploy and consume than other implementations. REST API returns data in one of two possible formats: Extensible Markup Language (XML) and JavaScript Object Notation (JSON) [12] which make open data easily consumable.

2.3 Open Data Protocol (OData)

OData (Open Data Protocol) is an ISO/IEC approved, OASIS standard that defines a set of best practices for building and consuming RESTful APIs. OData was first published by Microsoft under the Open Specification Promise. It is a standard that allows the creation of REST-based data services where resources can be managed by CRUD operations using simple HTTP messages. The protocol enables the creation and consumption of REST APIs, which allow Web clients to publish and edit resources, identified using URLs and defined in a data model, using simple HTTP messages. OData shares some similarities with JDBC and ODBC; like ODBC, OData is not limited to relational databases [5]. OData is based on several Internet standards (from bottom to top): HTTP, XML, Atom and AtomPub. Atom (the Atom Syndication Format) is the XML format through which OData publishes or sends data. It can also use JSON to send or publish data. AtomPub (the Atom Publishing Protocol) is a protocol for publishing and editing the Atom data that OData relies on. OData provides a query language directly in the URL to get data from the service. It supports HTTP methods- (GET, POST, PUT and DELETE) [1]. OData is intended to be used to expose and access information from a variety of sources including, but not limited to, relational databases, file systems, content management systems, and traditional Web sites.

2.3.1 OData Metadata: The OData service provides a service document describing the collections exposed. It is located at the root URI of the service. Performing the operation $metadata on the OData service retrieves the metadata document describing the EDM (Entity Data Model). The EDM formally describes the properties of the exposed resources and the central concepts are entities, relationships, entity sets and functions. The metadata document is represented in the XML-based CSDL (Conceptual Schema Definition Language) [1].

2.3.2 OData Queries: The result of a query is by default in the Atom format, but results can also be retrieved in XML or JSON format by using $format. There are several client libraries available on different platforms to facilitate consuming OData including Microsoft .NET Framework 4.0, Java, JavaScript, PHP and Excel 2010 PowerPivot. Server-side implementations for producing OData includes Microsoft .NET Framework 4.0 and IBM WebSphere [1].

2.4 Open Government Data Stage Model

Open Government Data is a new concept which is believed to be an extension of e-Government. However, the existing e-Government stage models are not capable of describing the increasing OGD movement [6]. The following models show the evolution of OGD with various models describing different stages to OGD maturity.

2.4.1 The Open Government Data stage model by Evangelos Kalampokis, Efthimios Tambouris, and Konstantinos Tarabanis: In order to supplement existing e-government stage models, ( ) proposed a new OGD stage model in their work

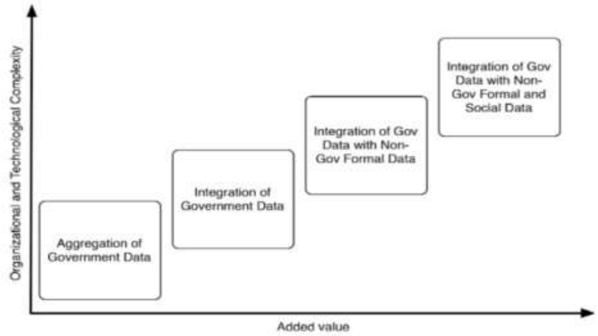

Figure 2: The Open Government Data stage model. (Source: E. Kalampokis, E. Tambouris, and K. Tarabanis).

after reviewing existing e-Government model aiming at (1) providing a roadmap for open government data re-use and (2) enabling evaluation of relevant initiatives [1]. The vertical axis presents the technological and organizational complexity that is involved in the provision of the data as open data gets matured while the horizontal axis presents the capability of developing added-value services based on the increased in the value of data being provided

2.4.2 The open government data maturity model by Joshua Tauberer: A more comprehensive maturity model to monitor OGD progress and ensure a path to sustainable development through OGD initiatives is proposed by Joshua Tauberer. The model provides a roadmap for deciding on many aspects of OGD [6]. The model is shown in the Figure 3.

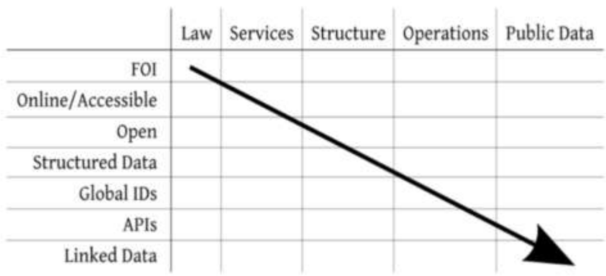

Figure 3 The open government data maturity model. (Source: Joshua Tauberer).

According to Joshua Tauberer, the model is read by starting at the top-left and going towards the bottom-right. Down the rows on the left side of the chart are the different technological strategies of open government data: freedom of information, using the Internet, principles of openness, structured data, global IDs, APIs, and linked data (the semantic web). Across the columns at the top are the different sorts of public information governments produce laws, service related data, operational data such as rule-making dockets and spending records, and finally a catch-all column for other public data (sometimes produced incidentally to government functions). The chart could be interpreted as a map of the open government data field.

2.4.3 Solar's maturity model for open data: In (Solar et al., 2012), the authors propose a maturity model for open data, with the aim of assessing the commitment and capabilities of public agencies in pursuing the principles and practices of open data. The authors extend the discussed guidelines and principles by considering other aspects towards publishing data, including an Establishment and Legal Perspective, a Technological Perspective, and finally a Citizen and Entrepreneurial Perspective which is the innovation and value creation derived from opening up government data [8].

2.*4.4 A five-star rating system for linked data:* A Linked data is currently considered the highest level of open data that encourages data reusability more effectively and efficiently. In 2010, Tim Berners Lee presented a five-star rating system to encourage the implementation of linked data [1]. Many governments in Africa have published data in various unstructured formats that are not promoting innovation. Getting to a three-star is critical to encouraging reusability by applications developers/entrepreneurs.

★	Open data with an open license published on the web in whatever format.
★ ★	Data presented in a machine-readable format (e.g. excel instead of an image).
★ ★ ★	Same as two stars but the data must be in a non-proprietary format (e.g. CSV instead of excel).
★ ★ ★ ★	All above but the data must be marked with URIs so that it can be pointed at.
★ ★ ★ ★ ★	All above but the data must be linked to other data to provide context.

Figure 4: A five-star rating system for linked data. The stars represent(s) the level of openness for data. (Source: Helena Lindén and Johan Stråle)

2.4.5 Open Government Data Life-Cycle: (Judie Attard et al, 2015) proposed a detailed open data life-cycle tailored to the specific needs of open government data. It covers all the processes in the life cycle of open government data in order to provide a standard process which government open data process can follow. The proposed life-cycle is made up of three sections namely, pre-processing, exploitation and maintenance sections [9].

The processes as defined in the life cycle ensures open government data is sustainable from preparing and publishing the data to maintaining the published data.

*2.4.6 Example of Open Data in Action***:** Examples of Open Data in action from distributed system perspective was presented by Helena Linde and Johan Strale in their research work titled "An Evaluation of Platforms for Open Government Data". The two examples presented were Midas and GovWild. These examples demonstrated a practical example of systems that use open government data in text-based formats such as HTML and XML as sources to create linked data. Linked data is a more advanced form of open data.

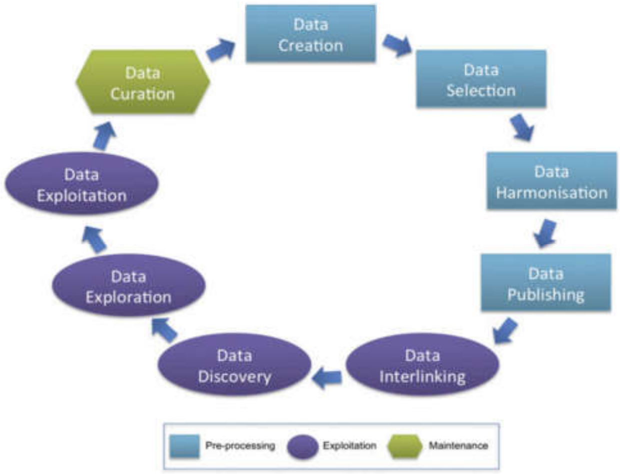

Figure 5: Open government data life cycle (Source: Judie Attard, Fabrizio Orlandi, Simon Scerri, Sören Auer)

*2.4.6.1 Midas***:** Midas is a scalable Hadoop-based system built in 2009 by employees at IBM. It is used for extracting, integrating and aggregating data from text or semi-structured regulatory financial filings. The filings originate from SEC (the United States Securities and Exchange Commission) and the FDIC (the Federal Deposit Insurance Corporation) and are available online as public information. The system is thought to be used by investors, financial analysts, lawyers and bankers. Use cases include potential investors who need to understand the web of relationships a company has with other companies and loan officers who need to understand inter-company relationships to estimate the total debt of the company and its subsidiaries.

*2.4.6.2 GovWild***:** GovWild (Government Web Data Integration for Linked Data) is a joint project between the Hasso Plattner Institute and IBM's Almaden Research Lab based on the Midas project. It structures and integrates open government data about politicians, companies and government funding. Like Midas, GovWild is based on Hadoop, Jaql and SystemT. The data sources used are from the US and EU, mainly Germany. The sources consist of online web content (HTML), text content (XML) and database dumps (CSV and TSV).

The graphical overview of the two examples are shown below.

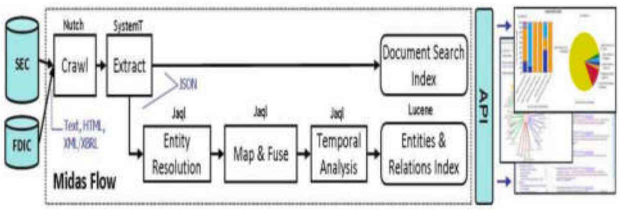

Figure 6: The data flow of the Midas platform where financial data from FDIC and SEC is transformed into linked data. (Source: Helena Linde and Johan Strale).

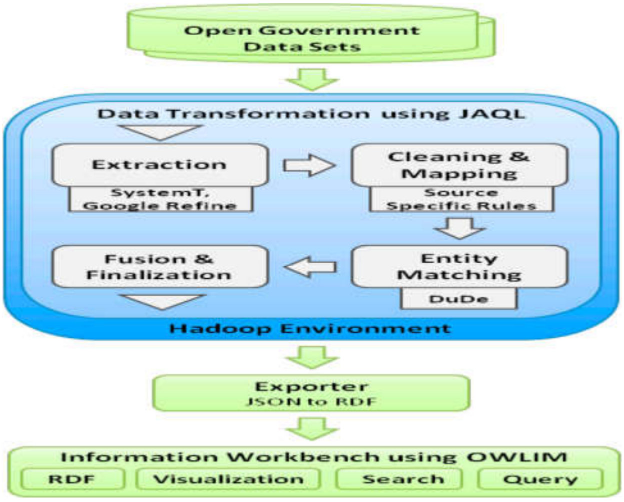

Figure 7: An overview of the architecture of the GovWild platform.

Table 1: An example of open source and commercial OGD Platform.

S/N	Open Source	Commercial
1	CKAN (the Comprehensive Knowledge Archive Network)	OpenDataSoft (ODS)
2	Open Source OGDI (Open Government Data Initiative)	Information Workbench
3	Open Government Platform (OG)	Junar
4	Liber	Data Visualization Wizard
5	GeoNetwork	Socrata

(Source: Helena Linde and Johan Strale).

2.5 OGD Cloud Computing Platforms

With much demand for open data recently, a number of cloud computing platforms have extended their services towards meeting this demand by way of providing suitable cloud hosting services for open data. Examples are Amazon Web Services, Microsoft Azure, Fi-ware Cloud, IBM, Google Cloud Platform, Oracle Cloud. These companies offer dynamic cloud computing platforms suitable for various open government data projects.

2.6 OGD Platform

There are several platforms for publishing and consuming open data; some are commercial while other are open source. Few examples of popular open data platforms are organized in table 1 as open and commercial sources respectively. They are mostly provided as a cloud service and support intensive data organization, transformation and integration to produce datasets as linked data. The common feature in all these platforms is API that enables data publishers and developers to interact with the open platform.

3 OPEN GOVERNMENT DATA CONCEPTUAL MODEL AND OPEN MIDDLEWARE ARCHITECTURE

3.1 OGD Conceptual Model

In order to make open data relevant to different data consumers and also satisfy various OGD guidelines and principles, there is need to holistically conceptualize a model that will best represent each group of consumer's interests. There are two major stakeholders of open government data: ordinary citizens who want the government to be transparent and accountable by having access to relevant public data and startups and/or enterprises either big or Micro Medium Small Enterprises (MSMEs) that are interested in public data for the creation of public value through innovative services to improve citizens' quality of lives. In order to fully take advantage of the above discussed technologies to accommodate the duo dimensions, more specifically, economic and social as well as transparency and accountability dimensions to open government data, the unstructured open government datasets in the form of government reports, circulars, special notices, policies, guidelines etc. and structured data from separate public data-driven applications have to be integrated into a national open government portal. The design of the portal is such that the primary/unstructured datasets are continually converted to standard data sets formats and integrated with standard datasets of government agencies' data from a centralized location to produce linked data. However, the unstructured data has to be presented to the first group for transparency and accountability purpose before it is converted to standardized data/datasets. The standard data sets from the two sources are integrated on the national open government portal through the OGD computing platforms for the second group. The data sets are grouped into sector specific. In consideration of a future and a highly matured OGD model, the following design considerations are recommended for the standard data while implementing National OGD portal:

- Open data/datasets (as defined in national OGD policy/law, or other law/policy in support of open government partnership OGP/OGD) from government agencies and non-government organizations who own public data in databases and information systems as well as smart initiative projects (IoT solutions) and social media data should be converted into standard formats (XML or JSON) in a centralized location before integration into the National OGD platform; and

- Data/datasets are presented to private businesses and startups in linked data format and make the data consumable through APIs on OGD portal.

If the national OGD portal is planned and designed in line with these value propositions, the economic and social benefits in terms of employment, new services creation, investment opportunities will be drastically increased. It is important to note that these propositions cannot be achieved without:

- the right laws/policies on OGD;
- open data enlightened and friendly institutions and political environments, multi-stakeholders' engagement and involvement;
- developing the right skill sets to take advantage of the standardized data the OGD portal offers; and
- creating awareness and educating ordinary citizens and other stakeholders on the potential possibilities and innovations that will emanate from OGD.

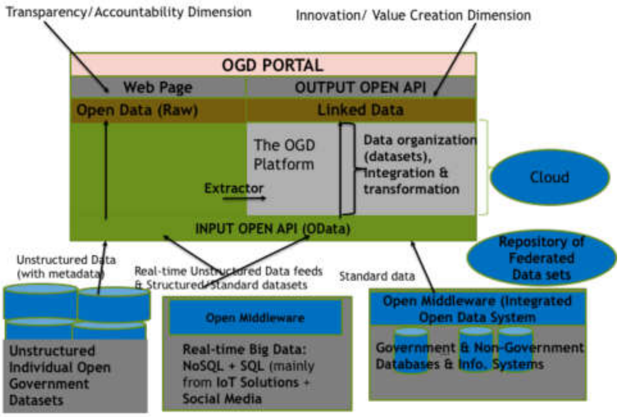

Figure 8: Two dimensional OGD model

These are critical to OGD sustainability. In addition to the design plan for structured and standard data, provision of unstructured/raw data should be part of the national OGD portal plan and design. This will ensure accessibility to information by ordinary citizens and other stakeholders who want to know about government activities for transparency and accountability purpose in fulfilling the first dimension to open government data. Also, having a one stop shop access to relational data of government agencies in datasets mostly by sectors before integration into OGD platforms is different from the traditional model of opening up data and a creative one and additional step that enhances how businesses and individuals access data for innovation and value creation with the overall of making OGD information system highly efficient and sustainable.

The conceptual model for a sustainable Open Government Data portal based on literature reviews is presented in figure 8 and explained as follows.

3.2 OGD Architecture

The main components of the architecture in the model is explained as follows:

National OGD Portal: for presenting both standardized (linked data) and unstructured data/datasets;

OGD Platform: for hosting massive unstructured and standard open government datasets. This is where complex data mapping/fusion, matching, conversion, integration and transformation take place for producing linked data/datasets. Due to the nature of data transformation that is expected to take place at the platform, it is usually compute-intensive by design and supports massive open datasets consumption through APIs. It is typically provided as a service to open data initiatives by cloud service providers or it could be built in-house;

Open Input and Output API: For consuming standard open data/datasets into and from the OGD platform. They specifically provide input and output access to standard datasets in the Open Middleware (Integrated Open Data System to be discussed in the next section) and OGD portal. They are integral part of the OGD platform and being provided as a service on the platform;

Open Middleware: provides a repository of federated and censored datasets from government agencies' databases and information systems as well as data from national IoT solutions and social media which are meant to be integrated into OGD platform; and

Data Extractor: for extracting text, HTML etc. of unstructured open government datasets in the form of government reports, circulars, special notices, policies etc. from OGD website (home/content page) which are to be integrated into the OGD platform with other datasets for conversion/transformation into linked data.

3.2 Architecture Scope

This paper limits its architecture scope specifically to Open Middleware for the government or non-government databases subscribed to OGD agenda. It describes the reference architecture of the open middleware but not the detailed implementation of the system architecture. In addition, the complete model which includes open middleware for big data stream is not part of this paper.

3.3 Description of Open Middleware for Government Databases and Information Systems.

This paper builds its foundation of OGD architecture from the idea of Open Middleware by Richard Pledereder, Vishu Krishnamurthy, Michael Gagnon and Mayank Vadordaria on "DB Integrator: Open Middleware for Data Access" [10] and other literature to provide a federated database system that suits Open Government Data needs and objectives. Governments all over the world as special institutions exhibit common characteristic when it comes to database integration and centralization. The common characteristic is the high tendency of government agencies in possession of public data to refuse to share in an integration fromat their data . This is an institutional problem that also needs some element of technological solution. However, the technology solution must be sustainable. This therefore necessitate the need for a Multi-database management system (MDBMS) in the form of a federated database system. A federated database system is suitable for solving this type of institutional problems where each component database of each agency exists as an independent entity. Each agency still owns and maintains its database but share part of it as agreed in the OGD policy/law on the OGD platform. OGD portal, platforms, federated database and local databases that appear as a single system are an example of a distributed system. The Open Middleware (federated database system) provides a middleware service between local databases implemented on different platforms (Network, Database Management Systems and operating systems etc.) and OGD platforms. Figure 9 depicts the federated database system/(MDBMS) as a middleware service provider between independent local DBs with different OS, distributed applications and end users' computers.

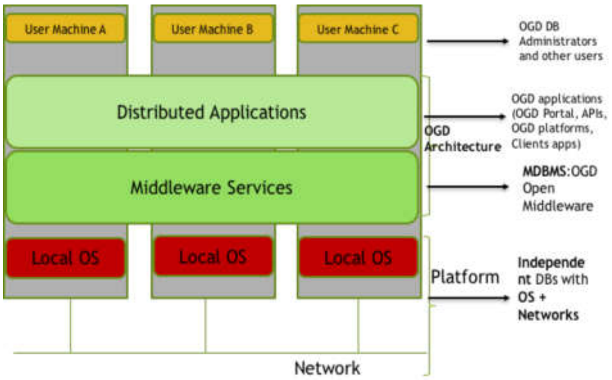

Figure 9: OGD Architecture as a Distributed System

3.4 Open Middleware as a Federated Database System

In a federated data system, individual source systems have control over their own data but agree to share some or all of this information with other participating systems upon request. The data are not stored by the system but rather imported [11] and can be uploaded through an API into the OGD platform to be organized, mapped, matched, transformed and integrated with other datasets. The individual sources of data store, secure, maintain the control of their data but release censored data based on OGD policy into the Open Middleware upon request by Open Middleware. The Open Middleware is likened to a federated database system which of course is a Multi-database management system (MDBMS).

The federated MDBMS provides users with a single system view of data distributed over a large number of heterogeneous databases or file system for legacy systems. It is typically composed of the following processing units:

- Language application programming interface (API) and SQL parser;
- Relational data systems (Global catalog manager etc.); and
- Gateways driver to access data sources [10]

3.5 Open Middleware as an Integrated Open Database System (IODS)

The federated MDBMS is referred to as Integrated Open Database System (IODS) in this paper. The (IODS) represents an open database. The IODS provides an open middleware service to the various databases of government Agencies who owns public data or an aggregated database for each sector of the economy as the case may be. It is a loosely coupled heterogeneous and federated multi-database system. The IODS interoperates with the individual source component databases similar to the way SQL in a relational DBMS interoperates with the record storage system.

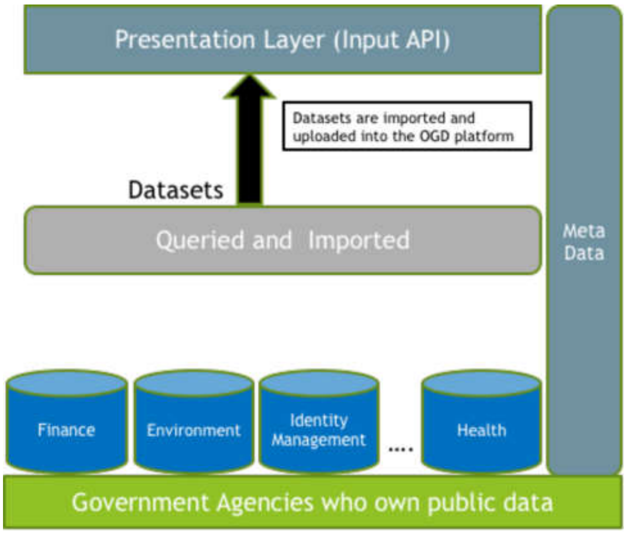

Figure 10: Open Middleware as a Federated Data Systems

The IODS has its own database and operates a global autonomous catalog management system. This catalog describes the data available in the multi-database. For data that resides in a relational database, the metadata definitions of table objects, index objects, and so forth, are imported (i.e., replicated) into the multi-database catalog. For data that resides in some other data source such as a record file system (e.g., record management system [RMS]) or a spreadsheet for legacy systems, the IODS catalog contains a relational description of the data source. This is the way metadata is organized. Metadata is defined as the attributes of the data that are accessible (e.g., naming, location, data types, or statistics [10]. The idea of cataloging ensures efficient search and preview of data. In addition, it provides context for the data in the IODS where data relationship with other data and lineage can be determined.

IODS has a database administrator whose job is to create and administer the integrated database and provides a one stop/shared intermediary interface for queries. The database administrators of OGD subscribed agencies either manually imports or the IODS itself automatically imports the open datasets into the Open Middleware (IODS). The IODS by design has access only to the designated tables from local databases of the government agencies that have subscribed to the OGD agenda. Precisely, it has access only to the vertically partitioned views (tables) specifically assigned to OGD requests in line with OGD law/policy. Since government databases are relational, the imported data are in the form of tables are then converted into standard file formats such as CSV, TSV, XML or JSON etc. These standard file formats are uploaded into the OGD platform. Automatic import is efficient, however, manual datasets imports might be required as a backup to carter for unusual incidents. The exposed datasets have metadata for proper description in line with open linked data standards.

In addition to this design consideration, the Open Middleware approach also emphasizes the following design directions:

- Three-tier architecture/integration model;
- Global, integrated catalog for data/metadata management; and
- Standard database adapter/gateway drivers (such as ODBC/JDBC database connectors) to access data sources [10].

The three-tier architecture data integration model allows for separation of functions thereby making IODS to access a very large number of databases including other IODS databases efficiently. Part of the integration is to allow IODS database provides datasets for specific sectors of the economy by enabling sectoral integration of two or more agencies. On that note, the architecture allows for seamless integration of different agencies and/or sub-IODS's databases for the provision of sectors specific datasets on the IODS. It is efficient to perform simple integration functions on IODS because most of the data mapping, matching, and integration are performed in the cloud on the OGD platform. The three-tier architecture of Open Middleware (IODS) comprises of:

- Web Client (including Open API);
- Open Data integrated catalog; and
- Local Databases (Sources).

3.6 The Description of the IODS Architecture

3.6.1 Web Client: The open Web client provides access to data and the metadata in the IODS server and as well interface with the OGD platform. It is SQL enabled and provides a user interface for OGD administrator to interact with OGD platform and IODS server. It is embedded with Open API which allows OGD platform to consume open datasets from IODS. There are open source REST API platforms for turning databases into an API platform. They provide an easy and secure way to add a REST API to any SQL database. The web client also provides OGD administrator the access to upload datasets into the OGD platform from IODS server. The implementation of the web client is based on OData because it contains all web protocols for open data.

3.6.2 Open Data Integrated Catalog: The Open Data integrated catalog is for data and metadata management of local data sources. The global catalog enables IODS to tell users (specifically IODS DB administrator, OGD platform, businesses and individuals) what data is available without requiring immediate connectivity to the local data sources. The clients do not need to deal with the complexity of the distributed data; IODS is managed and maintained as an independent database. It ensures importation of data and metadata definitions of the underlying tables of data sources.

This enables IODS to provide web clients with a single access point to the federated database environment. It can be implemented using any vendor data manager (MySQL, Microsoft SQL, Oracle etc.). Because of a single point of failure exhibits by IODS, standard high-availability mechanisms is employed to eliminate its potential failure within a node, a disk, or a network. This may involve clustering the nodes and replication of IODS catalog tables. An Open Data integrated catalog consists of a set of tables that IODS creates to maintain the IODS metadata (also referred to as the catalog) and the distributed open data that is available when connected to the IODS catalog [10].

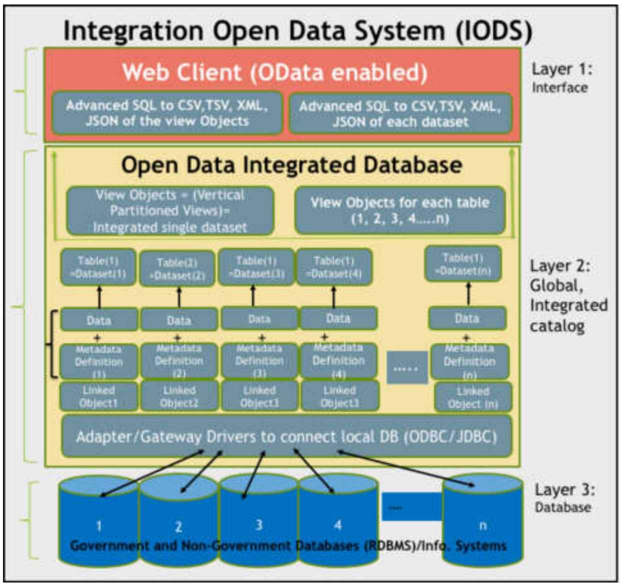

Figure 11: IODS Three-tier Architecture Approach

In addition to regular SQL objects such as tables or columns, IODS supports and uses links and proxy objects that are outside the scope of the SQL language standard for each data source secure connection to IODS [10].

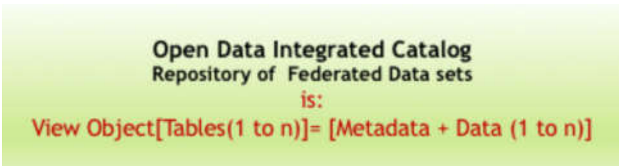

Figure 12: Open Middleware (IODS) Components Structure

3.6.2.1 Links and Proxy Objects: The link object tells underlying local databases how to connect to IODS (referred to as the link database). A link object has three components: a link name, the access string used to attach to the link database, and, optionally, security information used by the IODS adapter/gateway driver to provide authentication information to the link database system. The proxy object is associated with a link object. It can be used to specify user-specific authentication information for individual links. When users do not want to use proxies for their links, they must specify the authentication information for a specific database at the time they connect to [10] IODS.

3.6.2.2 Tables: With link and proxy objects in place, the users (local administrators) can import data and metadata definitions of underlying tables into the IODS catalog. The metadata imported for a table describes the data in the table. The import step is performed with a statement that allows for a link reference [10]. The tables allow the IODS administrators to have access to tables containing data and metadata from the local databases. The data with metadata in the table is converted to standard datasets by web client. The dataset is uploaded into the OGD portal.

3.6.2.3 View Objects: View objects are useful for making multiple tables from different link databases appear as a single table. In IODS, views serve as powerful mechanisms to resolve semantic differences in tables from disparate databases. It allows IODS to present the entire tables as a single table (containing data and metadata) to the OGD platform through the web client at a particular time. The dataset can be converted to standard formats as presented in figure 11.

3.6.2.4 Gateway/Database Driver: Gateway drivers enable access to specific data sources and data from legacy systems. The combination of gateway drivers, API and SQL parser allows importation of data from local sources.

3.6.3 Local Data Source: The local data sources are the various databases of government and non-government organizations that allow their censored data to be imported into the IODS. This also includes data from legacy systems. The IODS implementation and the roles of local data sources are stated in the OGD policy/law.

3.7 Open Middleware Security: The security of the local databases and legacy systems is the primary responsibility of the individual agencies or sector-specific IODS. Access to each local data source by the IODS administrator(s) or local database administrators access to the IODS is highly protected and secured through any PKI-enabled credentials, strong authentication provided by National PKI or any other acceptable security means specified in the OGD policy/law. The OGD Database Administrator privilege is assigned to users responsible for setting up and maintaining an IODS database. De-identification should be the primary responsibility of data source Agency.

4 CONCLUSIONS

The paper presents a model and architecture for Open Government Data portal having considered the needs of the major OGD stakeholders in Nigeria. Access to relevant public data in various formats that best represents each stakeholder's interest is critical to the success of Open Government agenda. Government transparency & accountability, as well as innovative, data-driven value creation are at the heart of OGD agenda.

Furthermore, the paper presents arrays of evolving technologies suitable for publishing, accessing and consuming open government data by different stakeholders. The model presented in the paper is unique in the sense that it accomplishes the objectives of the two dimensions of open government data i.e. transparency and accountability as well as value creation in terms of social and economic values.

The major contribution of the paper is the introduction of IODS for solving issues around data ownership among government agencies. The IODS promotes data quality, discovery, security and privacy by design and helps build trust towards opening up data by government agencies. In addition, the separation of a centralized or one stop shop relational open data from the OGD portal is a drastic deviation from the traditional method of opening up data. This enhances the performance of the entire system and allows open up of APIs to relational data of government agencies who own public data and/or sector specific relational data for data users in a central location. Data users do not need to go through multiple agencies' APIs. Therefore, the architecture is aimed at avoiding direct integration and a high API overload on individual source databases. The developers/entrepreneurs can get datasets of choice, load it into their own applications and design their APIs according to their use case locally on their IT infrastructure.

Ultimately, IODS ensures sustainability of OGD by allowing government agencies release non-sensitive data (according to the OGD policy/law) but still responsible for ownership, overseeing, maintenance and security of their source databases.

The model and architecture are being used as a guide for the implementation of Nigerian Open Government Partnership Portal and if successful, the experience and lesson learnt will be shared with other African countries and other countries who are yet to launch OGD portal for adaptation and technology transfer.

The future research will look into developing IODS for real-time data (from IoT solutions and social media) integration into the OGD portal and how best big data analytics can be used to create new opportunities for businesses, developers/entrepreneurs in deriving new values from the combination of the integrated single dataset and the extracted data from government's reports, circulars, policies, special notices etc. available on the OGD portal.

It is hoped that this work has initiated OGD model and architecture for more research toward a widely accepted architecture as well as added to the literature works for its contributing to the body of knowledge.

A APPENDICES

A.1 Requirement Consideration for Federated Database System Characteristics

These considerations are modified excerpt from Centralized vs. Federated: State Approaches to P-20W Data Systems by the national center for education statistics (Institute of education sciences) [9] to suit the concept of OGD.

Table 2: Requirement consideration for Federated Database System

S/N	Requirement Consideration	Characteristics
1	Data Ownership	Data ownership is the source agency but share a censored copy of it at a given time in line with OGD policy/law
2	Staff Resources	Staff Resources are required of each source system to oversee and maintain required data access. In addition, there would be a need for technical support for data extract, transform and load (ETL) processes of de-duplication at the data sources. Staff resources are required from each participating agency to review, approve and maintain data requests based on OGD policy/law.
3	Technical Requirements	Each source system needs the required hardware and network bandwidth to facilitate and process IODS requests/ queries (ETL tools), conduct de-identification and returning the resulting dataset.
4	Privacy/Security	Primary responsibility for privacy/security is with the data source agencies. The secure process needed for handling database requests/queries and de-identification should be the primary responsibility of source agency in line with the OGD policy/law.
5	Update/Corrections	Data resides within each agency. Each Agency is responsible for updating the local databases/sources.
6	Data availability	Access to data is determined in the source agency. The source agency should make data available IODS.
7	Scalability & Interoperability	The IODS plan and design is based on open standards to ensure the system resources are interoperable and scalable. Seamless addition of resources and local data sources into the system without hampering system performance is at the heart of IODS design.
	Sustainability	A centralized funding allocated for IODS or each agency to make their contribution based on funding formula for the support of the processes needed for establishing and maintaining federated IODS system.

ACKNOWLEDGMENTS

The authors would like to acknowledge the efforts of Prof. Eunmi Choi of the Business IT department, Kookmin University, for her excellent contributions and mentorship to Mr. Lukman Lamid during his Masters Programme in e-Government. In particular, her Distributed Systems class was instrumental in understanding the different suitable technologies and architectures for deploying e-government systems, the foundation upon which this paper is based.

REFERENCES

[1] Helena Linden and Johan Strale. An Evaluation of Platforms for Open Government Data. KTH (Royal Institute of Technology) School of Technology and Health 136 40 Handen, Sweden, 2015

[2] George Coulouris, Jean Dollimore and Gordon Blair: Distributed System – Concepts and Design, Fifth Edition. 2012

[3] Quora. https://www.quora.com/Markup-Languages-What-are-the-advantages-of-JSON-over-XML. (last date visited, November 19, 2016).

[4] Open Data Handbook. http://opendatahandbook.org/guide/en/appendices/file-formats/ (last date visited, November 19, 2016).

[5] Wikipedia, https://en.wikipedia.org/wiki/Open_Data_Protocol. (Last day visited September 16, 2017

[6] Evangelos Kalampokis, Efthimios Tambouris, and Konstantinos Tarabanis. Open Government Data: A Stage Model. M. Janssen et al. (Eds.): EGOV 2011, LNCS 6846, pp. 235–246, 2011.

[7] Joshua Tauberer. A Maturity Model for Prioritizing Open Government Data. https://razor.occams.info/pubdocs/ogdmatmodel.html;

[8] Solar, M., Concha, G., & Meijueiro, L. (2012). A model to assess open government data in public agencies. In H.J. Scholl, M. Janssen, M. Wimmer, C.E. Moe, & L.S. Flak (Eds.), EGOV. Lecture Notes in Computer Science. 7443. (pp.

[9] Judie Attard , Fabrizio Orlandi, Simon Scerri, Sören Auer. A systematic review of open government data initiatives, University of Bonn, Regina-Pacis-Weg 3, 53113 Bonn, Germany. Government Information Quarterly. August, 2015.

[10] Richard Pledereder, Vishu Krishnamurthy, Michael Gagnon and Mayank Vadodaria. DB Integrator: Open Middleware for Data Access, Digital Technical Journal Vol. 7 No. 1 1995.

[11] National Center for Education Statistics (Institute of Education Sciences). Centralized vs. Federated: State Approaches to P-20W Data Systems.

[12] Project Open Data: https://project-open-data.cio.gov/api-basics/ (Last visited December, 26, 2017)