electronics

MDPI

Article

SODAS: Smart Open Data as a Service for Improving Interconnectivity and Data Usability

Heesun Won[1], Jiwoo Han [1], Myeong-Seon Gil [2] and Yang-Sae Moon [2,*]

[1] CybreBrain Section, Future & Basic Technology Research Devision, ETRI, 218, Gajeong-ro, Yuseong-gu, Daejeon 34129, Republic of Korea
[2] Department of Computer Science and Engineering, Kangwon National University, 1, Kangwondaehak-gil, Chuncheon-si 24341, Republic of Korea
* Correspondence: ysmoon@kangwon.ac.kr

Abstract: In this study, we proposed Smart Open Data as a Service (SODAS) as a new open data platform based on the international standards Data Catalog Vocabulary (DCAT) and Comprehensive Knowledge Archive Network (CKAN) to facilitate the release and sharing of data. We first analyze the five problems in the legacy CKAN and then draw up corresponding solutions through three core strategies: CKAN expansion, DCATv2 support, and extendable DataMap. We then define four components and nine function blocks of SODAS for each core strategy. As a result, SODAS drives Open Data Portal, Open Data Reference Model, DataMap Publisher, and Analytics and Development Environment (ADE) Provisioning for connecting the defined function blocks. We confirm that each function works correctly through the SODAS Web portal, and then we apply SODAS to actual data distribution sites to prove its efficiency and practical use. SODAS is the first open data platform that provides secure interoperability between heterogeneous platforms based on international standards, and it enables domain-free data management with flexible metadata.

Keywords: open data; SODAS; data hub; data distribution; data sharing; open data platform

check for updates

Citation: Won, H.; Han, J.; Gil, M.-S.; Moon, Y.-S. SODAS: Smart Open Data as a Service for Improving Interconnectivity and Data Usability. *Electronics* **2023**, *12*, 1237. https://doi.org/10.3390/electronics12051237

Academic Editors: Ivan Ganchev, Máirtín O'Droma and Zhanlin Ji

Received: 27 January 2023
Revised: 1 March 2023
Accepted: 2 March 2023
Published: 4 March 2023

1. Introduction

With the ever-growing applications of big data and artificial intelligence, the quality of the service is directly related to the quality of the data. Accordingly, various research, public, and industrial fields are investing a lot of effort to get high-quality data. Globally, however, data with a practical application value has been significantly lacking. In some countries, public data release is being facilitated by the government, but its actual use remains insufficient. This is because the data being released are often statistically summarized or low quality. To solve these problems, this study standardizes the quality management system to ensure data quality and proposes a standard technique for storing, managing and sharing high-quality data. In addition, to ensure the value and improve the utilization of the collected data, we propose "Smart Open Data As a Service (SODAS)", a novel open data management platform that enables efficient expansion and linkage among various open platforms.

SODAS provides not only existing open data but also collection and management functions for Internet of Things (IoT) data in a distributed environment. Therefore, SODAS can be used in IoT fields such as smart cities, smart energy, and healthcare, universally, in contrast to existing platforms that focus on sharing public data.

Representative platforms used for data sharing and distribution include Comprehensive Knowledge Archive Network (CKAN) [1], Open Government Platform (OGPL) [2], and Socrata [3]. CKAN is an open data platform developed by Open Knowledge Foundation (OKF) and is widely used in more than 40 countries, including the UK, the US, and Canada. Besides basic functions such as data registration, publication, and statistical analysis, CKAN provides extended functions of visualization, Application Programming Interface (API) management as plug-ins that can be combined with open sources.

In this study, we analyzed CKAN version 2.0 as a legacy CKAN and noted the following five problems.

Problem 1. *Data management limitation: Because of the functional limitations of CKAN, additional extension plug-ins installation and management jobs are required.*

Problem 2. *No real-time feature: As CKAN and the extension plug-ins do not support real-time data collection and management, there are restrictions on the data domains and formats that can be shared.*

Problem 3. *Lack of metadata: CKAN uses Data Catalog Vocabulary (DCAT) [4] as a data catalog standard. However, CKAN does not exploit the latest DCAT version, and thus limits the metadata that can be defined.*

Problem 4. *No interconnection standard: The existing platform has no metadata management guides for data publication and distribution, thus reducing the interoperability between open data platforms.*

Problem 5. *Low utilization: CKAN, which primarily focuses on data management and retrieval, has limited applications.*

To solve these five problems of the legacy CKAN, we design and develop detailed functions of SODAS based on three core strategies: *CKAN expansion*, *DCATv2 support*, and *extendable DataMap*. First, CKAN expansion is a core strategy to solve Problems 1, 2, 4, and 5. To overcome the functional limitations of CKAN, we analyze the necessary functions of the open data hub and fully expand it based on CKAN. Second, DCATv2 support is a core strategy to solve Problems 3 and 4 by improving data and service quality and defining a metadata management system. Third, extendable DataMap is a core strategy to solve Problems 4 and 5 by improving interoperability between platforms and supporting high-level data search and query functions. We discuss these five problems and core strategies in detail in Section 3.

Based on these five problems and core strategies, we then introduce four components of SODAS and design/implement the final platform based on these components. The four components are Open Data Portal, Open Data Reference Model, DataMap Publisher, and Analytics and Development Environment (ADE) Provisioning, respectively. Table 1 summarizes relationships among defined problems, core strategies, and SODAS components. We use these relationships to implement SODAS components as nine function blocks inside the platform. Each block consists of several functions necessary for open data sharing and utilization and supports 604 REST APIs to increase the usability of SODAS.

Table 1. Relationships among the problems, core strategies, and SODAS components.

SODAS Component	Related Problems	Solutions (Strategies)
Open Data Portal	Problems 1, 2, 4, 5	CKAN expansion
Open Data Reference Model	Problems 3, 4	DCATv2 support, extendable DataMap
DataMap Publisher	Problems 3, 4, 5	CKAN expansion, extendable DataMap
ADE Provisioning	Problems 1, 5	CKAN expansion

The first component, Open Data Portal, is configured by analyzing the legacy CKAN, optimizing each function, and expanding essential functions for data distribution management. We implement this component as a Web portal that provides overall functions of data management, distribution, and utilization based on the CKAN expansion strategy. The portal's main functions include authentication and authorization for user and institution management, multitenant support, data harvesting, and platform management. The second component, Open Data Reference Model, effectively manages and utilizes structured and unstructured data, and improves interoperability with other platforms to build a Linked Open Data (LOD) environment. We design this component based on DCATv2 support and extendable DataMap, and its main functions include DCATv2-based data definition standard management and data quality management through exploiting the metadata. The third component, DataMap Publisher, improves the essential extension plug-in Harvest of CKAN, through CKAN expansion and extendable DataMap, and it supports harvesting between heterogeneous platforms and real-time data collection and registration. This component enables efficient linkage of standard/non-standard metadata through a conversion tool that maps data catalogs of heterogeneous systems to DCAT. The fourth component, ADE Provisioning, is composed of user analysis tools and service extension functions. The cloud-based user analysis tool enables user-defined algorithms to be directly implemented and executed in an open-source-based cloud environment on SODAS. It also improves API utilization by providing a deployment automation environment that can implement and deploy services of SODAS through the service extension function.

The contributions of this study are as follows: First, we highlight the problems in the existing open data platform based on CKAN and propose a new open data platform, SODAS, by introducing three core strategies. Second, to ensure connectivity and scalability among open data platforms, we define extendable DataMap based on the latest international standard, DCATv2, and make domain-free data linkage possible. In particular, SODAS Harvester can more easily collect data and metadata from other open data platforms through DCATv2-based Open Data Reference Model. Third, we define four components and nine function blocks of SODAS to expand the functions of CKAN, and significantly improve the utilization of the open data hub by providing the REST API for each function. Fourth, we build an integrated platform based on the latest open-source projects to verify each function and demonstrate the effectiveness of SODAS through experiments that measure dataset collection efficiency and related dataset detection rates. Fifth, we apply SODAS to actual industrial sites, including KDATA, Financial Security Institute, Patient-care Advancement with Responsive Technologies and Engagement Together (PARTNER) project [5], and Korea Culture Information Service Agency, which proves the utilizability of SODAS.

The rest of this paper is organized as follows. Section 2 explains the related technologies, open data platform, and DCAT. Section 3 details the problems of CKAN and presents the core strategies for solving these problems. Section 4 introduces SODAS, a novel open data platform based on these core strategies. Section 5 verifies each function based on the SODAS portal to prove the effectiveness of SODAS. Finally, Section 6 concludes the paper.

2. Related Work

2.1. Open Data Platform

Open data can be freely used and can be redistributed without restrictions [6–12]. In general, open data needs to satisfy the following three conditions [13]. First, the data must be available and accessible. Second, the data must be reusable and able to be redistributed without restrictions. Third, universal participation through the data must be possible. Representative open data platforms include CKAN [1,14] and OGPL [2] as open sources, and Socrata [3] and Junar [15] are commercial platforms.

OGPL is an open data platform released in 2013 and developed by the governments of the United States and India. OGPL provides basic functions that are required for portal services of governmental agencies, but its usage status has been very low, as compared to CKAN. Socrata, a commercial software, is a cloud-based open data platform developed in the United States in 2007. Socrata provides many functions compared to open-source data platforms and has been used by data portals of US state governments. Junar, a commercial software similar to Socrata, was developed in Silicon Valley in 2010 and is used by certain metropolitan agencies in the US. This commercial software may provide more functions than other open-source software. However, they have critical problems including excessive maintenance costs and low interoperability, as compared to open-source platforms. Table 2 summarizes the pros and cons of these platforms.

Table 2. Pros and cons of existing open data platforms.

Name (Release)	Pros.	Cons.
CKAN (2007)	Open-source; Frequently used	Lack of operational/functional stability; Non-standard
OGPL (2013)	Open-source; Suitable for government	Infrequently used (only used in India); Non-standard
Socrata (2007), Junar (2010)	Supports various functions compared to open-source; Providing customized functions	High installation cost (commercial); Low scalability; Non-standard

CKAN uses PostgreSQL [16] for its database, SQLAlchemy [17] for its Object Relational Mapping (ORM) [18], and Apache Solr [19] for its search engine. CKAN solves the problems of insufficient functions by utilizing plug-ins, also referred to as an extension. Figure 1 illustrates the structural diagram presenting the basic functions of CKAN. CKAN basically supports metadata-based data registration and retrieval as its main functions without any extensions. It also includes a simple visualization that displays data information on a map.

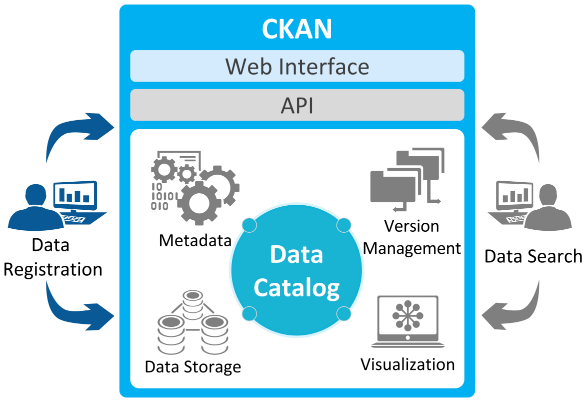


Figure 1. Basic operations of CKAN.

CKAN users are classified as platform administrators, data providers, and general users (who do not register data). The platform manager maintains the data catalog, which contains the core information of CKAN. The data catalog refers to the information composed of a list of datasets for searching and using data and metadata. The platform manager also determines and manages the main functions of CKAN, such as metadata management, dataset version management, and visualizations based on this data catalog. Data providers are mainly governments or organizations, and they publish data held by or collected from external sources through the Web UI and APIs. The process of collecting data from external sources is called harvesting [1], and this function requires an additional plug-in. Data users can search and download the data published by the data provider using the Web UI. Since several extension plug-ins are developed separately from the basic CKAN platform, compatibility problems between the main program and installed plug-ins often occur.

2.2. DCAT

DCAT [4] is a standard vocabulary defined by World Wide Web Consortium (W3C) to express catalog models and data attributes and is designed to facilitate the interoperability between data catalogs. CKAN provides a harvesting function that distributes, searches, and collects data catalogs based on the DCATv1 standard. Currently, the schema.org [20] community, led by Google, Microsoft, and Yahoo, is also establishing standard vocabularies for describing datasets. Google has been operating its dataset search engine [21] since 2018. In addition, schema.org manages datasets based on DCAT and provides mapping information with other standard vocabularies. Therefore, if the open data platform provides data information that is compatible with active standards, such as DCAT and schema.org, users can efficiently find the desired data in a wide, well-connected LOD environment. DCATv1, released in 2014, has been used by public portals around the world, including the UK, the US, and Australia. Subsequently, due to various issues such as low interoperability between different profiles, a lack of data quality, and historical information management, the W3C Data Exchange Working Group (DXWG) began revising the DCAT standard in 2017. As shown in Figure 2, they released the draft of DCATv2 in November 2019, which reflected various domain requirements.

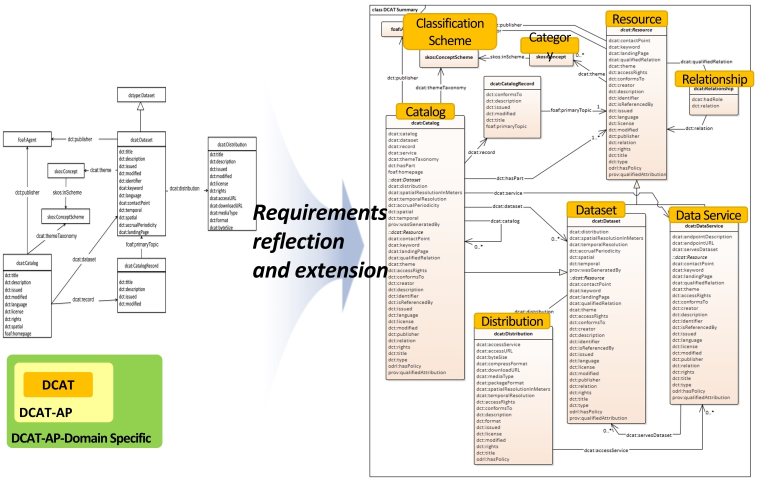


Figure 2. Structure comparison of DCATv1 and DCATv2.

DCATv2 reflected the essential elements necessary for data distribution and utilization by providing the scalability of the standard model. The main features are (1) the separation of OpenAPI, search, and queries included in the dcat:Dataset in DCATv1 in dcat:DataServices; (2) allowing dcat:Dataset and dcat:DataServices to inherit properties of newly defined dcat:Resource classes; (3) allowing various attributes created by inheriting the dcat:Resource class to be included in the data catalog; (4) the relationship between resources required to be expressed for a data search; and (5) providing a resource license and access authority documentation guide.

3. Problem Analysis and Core Strategies

In this section, we present five problems of the legacy CKAN and highlight three core strategies to solve these problems. In particular, we focus on the limitations of expandability and usability as a result of the legacy CKAN. Here, we present each problem in detail.

Problem 1, *data management limitation*, refers to the inconvenience and low scalability of construction due to the functional limitations of CKAN. As CKAN has only limited functions, to build an actual data-sharing system, we had to install many extension plug-ins such as Datastore, Datapusher, Harvest, and DCAT. CKAN's plug-in mechanism has the advantage of being flexible in selecting the functions required for the system but also has the disadvantage of requiring unnecessary additional jobs when building the system. Another critical problem is that essential functions for data management such as data collection and storage cannot be used without a plug-in. It has another critical problem that extension plug-ins developed by the third party cannot reflect CKAN's recent update immediately. Therefore, we need a new open data platform that supports flexible scalability while reinforcing the core functions of CKAN.

Problem 2, *no real-time feature*, refers to the lack of real-time data collection and management supported by CKAN, which has limited the provision of application domains and data types. Most open data that has been shared are file data composed of statistics for a specific period. Typical examples include the population distribution in a specific city, monthly precipitation, and monthly market trends. In the fields of big data and artificial intelligence, we generally handle hundreds of megabytes or more of text, images, and videos, so documented statistical data is not sufficient. Furthermore, the legacy CKAN cannot provide stream data for real-time services such as IoT, smart city, and autonomous driving. After all, to achieve the high utilization of open data, we improve CKAN to collect, accumulate, and provide real-time sensors, logs, and image data of various domains.

Problem 3, *lack of metadata*, means that CKAN cannot guarantee overall service quality in data management, distribution, and searches due to the lack of definable metadata. CKAN is basically configured based on DCATv1, and DCATv2 has yet to be actively supported. DCATv1 has many limitations in terms of describing the data characteristics because there are few types of metadata defined. Eventually, users cannot understand data characteristics using the metadata, and they need to open actual files and check the data to know the characteristics. In addition, it incurs limitations of metadata-based search and restriction of interworking with DCATv2-based platforms. As a result, Problem 3 seriously degrades the service quality of the overall data management platform CKAN.

Problem 4, *no interconnection standard*, means that CKAN has an interoperability problem with other platforms owing to operational errors and no metadata standard. As in Problem 3, CKAN cannot manage DCATv2-based information even if it had collected such metadata from other platforms and it did not support nonstandardized metadata. For example, the word “title” used instead of “subject”, “name”, “subj” in a data scheme makes data connection with other platforms very complicated. Since the data category is expressed in a tree structure, data connection between platforms becomes more complex without standards to express the structure. These problems reduce the interoperability with other open data platforms and eventually hinder the activation of the open data ecosystem centered on data sharing and distribution.

Problem 5, *low utilization*, refers to the inability of users to fully exploit the data because CKAN does not provide functions for non-expert data users. Even though governments are actively attempting to publish data, legacy platforms do not provide user-centered data visualization, search, and analysis, resulting in low utilization. As compared to data providers (or expert users), most of the data users are not experts in each domain, and thus, they need to put in a lot of effort to find the desired data from the public data. In addition, users need to build their own data processing, analysis, and service environments. These limitations cause the low utilization problem.

To solve these five problems, we present three core strategies for building SODAS. Each strategy is related to solving multiple problems, as shown in Table 1. Furthermore, the proposed core strategies are the main goals of the SODAS design.

The first strategy is *CKAN expansion*. This strategy aims to stabilize CKAN operations and to ensure essential functions as a data sharing and distribution platform. In this paper, we extensively analyze the operating process and source codes of CKAN and propose a novel open data platform from the main functions of CKAN. More specifically, we intend to solve Problems 1, 2, 4, and 5 by integrating data collection, storage, and management functions in the SODAS platform, in addition to including new functions such as data sales, purchase, and analysis as well as the and development tools for integrating SODAS into other systems.

The second strategy is *DCATv2 support*. This strategy guarantees the diversity of data expression and stable interoperability with other platforms by reflecting the latest catalog publication standards. Currently, most open data platforms manage data with DCATv1, and their main services are only list-based inquiries or simple searches. Furthermore, data analysts and developers are hard-to-understand data descriptions and access data easily. In this study, we aim to solve Problems 3 and 4 by explaining data in detail so that non-experts can easily understand through DCATv2 support, and improve data connectivity between platforms based on current international standards.

The third strategy is *extendable DataMap*. Similar to DCATv2 support, this strategy ensures the diversity of data representation and stable interoperability between systems, while also providing generality covering standard/non-standard data catalogs. Extendable DataMap is a general-purpose data catalog that includes additional information such as dataset properties, relationships, and domain characteristics based on taxonomy. Figure 3 shows an example of profile management through the classification system and the category of extendable DataMap. As shown in the figure, the highest category defines a management scheme that applies DCATv2, and lower categories can inherit the scheme and add extend necessary metadata. SODAS creates this extended catalog as a Resource Description Framework (RDF) file expressed in DCATv2. Thus, SODAS can easily exchange data information with other platforms such as Socrata and Junar as well as CKAN. It can also provide various types of data search and analysis functions. Consequently, Problems 4 and 5 can be solved.

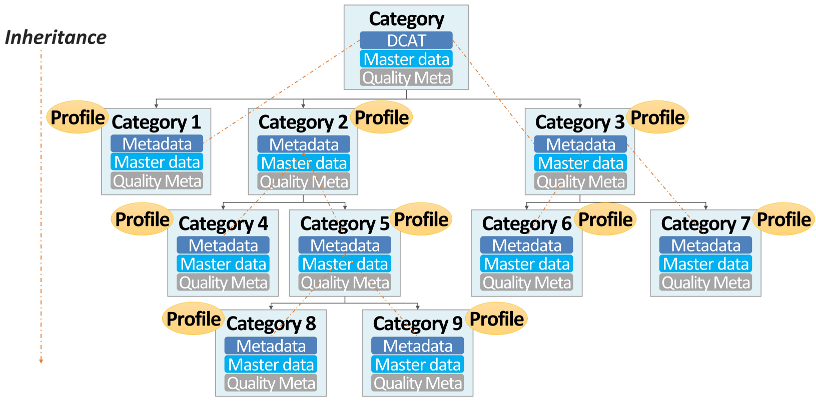


Figure 3. An example of DCATv2-based profile management.

The problems and core strategies described thus far have become the goals for the functional design of SODAS as presented in this study. In Section 4, we present the overall structure of the SODAS platform designed to meet these goals and explain in detail the components and function blocks of SODAS required for the data sharing and distribution platform.

4. SODAS Framework

4.1. Overall Framework

This section briefly describes the entire framework, components, and function blocks of SODAS designed by the core strategies. Figure 4 shows the overall framework of SODAS, composed of components and function blocks. In the actual service, each function block is organically connected to run Web interface-based components. In the figure, SODAS consists of three legacy blocks (dotted lines) that update the functions of the legacy CKAN and six new blocks (solid lines) that are newly added. We design each function block based on open-source and integrate several open-source frameworks through Node.js. Table 3 summarizes the main functions of each block. SODAS provides the main functions of each block as a REST API for the convenience of users' service development.

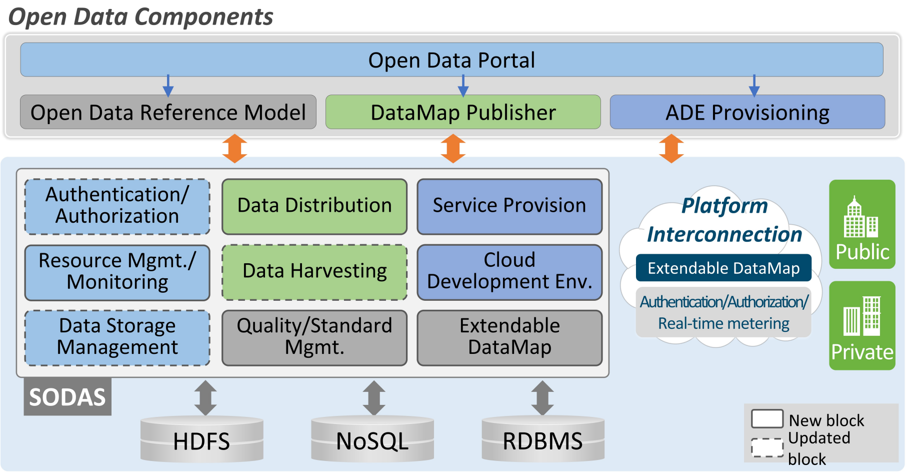


Figure 4. The overall framework of SODAS.

Table 3. SODAS function blocks and main actions.

Block Name	Main Actions
Data Distribution	Data registration/modification/deletion/purchase function, data landing page management
Authentication/Authorization	Single Sign On (SSO) management, organization/user role management, authorization management according to data purchase
Quality/Standards Mgmt.	Classification system/category/metadata management, quality verification rules, quality measurement & history management
Data Harvesting	Open data harvesting, real-time data (IoT) harvesting
Extendable DataMap	DCATv2-based catalog sharing & distribution, metadata conversion tool
Cloud Development Env.	Analysis/development environment (Theia, Notebook, TensorFlow, etc.), sandbox resource management and provisioning
Data Storage Mgmt.	HDFS/HBase/MongoDB/Jena/RDBMS integrated management, capacity control by organization & user
Resource Mgmt./Monitoring	Computing resource allocation and monitoring by tenants(organization and uses)
Service Provision	Algorithm & service distribution, currently for dataset search

The advantages of SODAS shown in Figure 4 and Table 3 are as follows. First, SODAS is designed with modularization-based block structures, so that users can apply only the necessary functions through its API. This means that the improved functions of SODAS can be easily incorporated into existing systems. Second, SODAS collects real-time streaming data, such as IoT data, with an improved Harvester, and stores the original data in its own storage system consisting of HDFS, NoSQL, and RDBMS. Therefore, it can be used without any problems, even if there is a failure in the connected data, such as service termination or file corruption. Third, users can perform pre-processing and data analysis using SODAS' ADE Provisioning without a separate system. Fourth, since all of these core functions are supported by the web-based Open Data Portal, an OS-free data distribution management service can be established by introducing SODAS.

Figure 5 shows the mapping of the core strategies, components, and function blocks with each other. As shown in the figure, four components and nine function blocks are defined around the core strategies, and each SODAS function operates around this connectivity. As shown in Figures 4 and 5, we derive necessary tasks for each component and define functional blocks by grouping these tasks. Therefore, the blocks of SODAS operate in connection with each other.

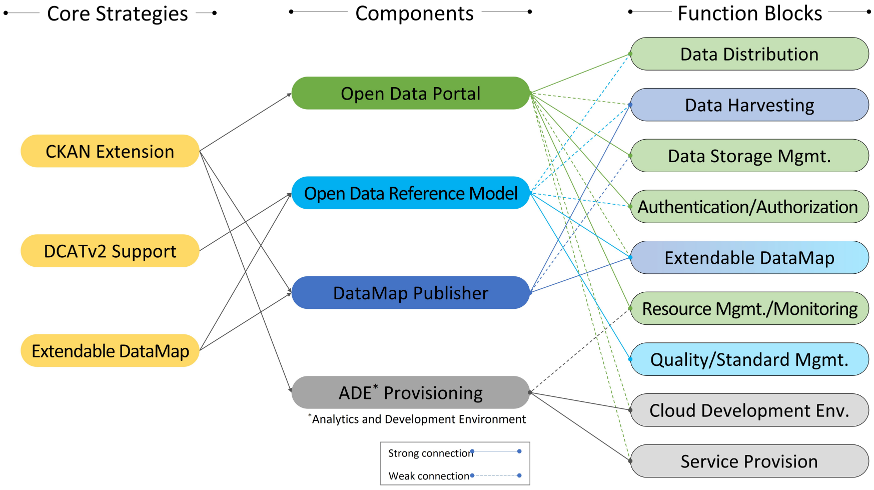


Figure 5. Relationship among SODAS major components and function blocks.

4.2. Open Data Portal

Open Data Portal is the largest component that connects all function blocks of SODAS based on the CKAN expansion strategy. Figure 6 shows the menu layout of the Open Data Portal. Portal users are classified into organizational users, individual users, and portal administrators (platform/system/standard administrators). They can use portal functions such as data registration, modification, deletion, purchase, and search according to the access authority specified in each menu. In particular, SODAS controls organization/user roles through multitenant-based authority management, and thus, it can set data publication and billing for each organization.

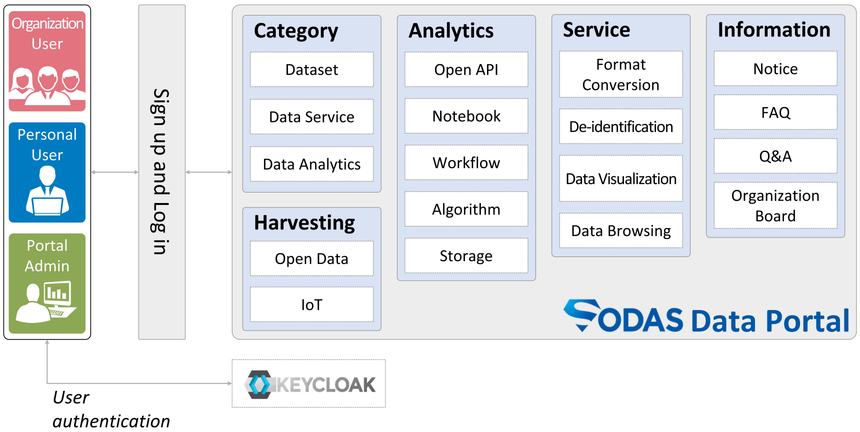


Figure 6. Menu layout of Open Data Portal.

A brief description of the main menus in the Open Data Portal is as follows. First, Category provides registration/search/purchase functions of datasets, data services, and data analytics algorithms. Users having datasets or algorithms can register each resource using the registration function. Each resource can be free or paid, and the user can add the desired resource to the shopping cart or favorites. Users can also directly analyze their own datasets in the Data Analytics menu and search for similar or related datasets via the Semantic Search sub-menu which uses the taxonomy information. Second, Harvesting basically provides the ability to collect data catalogs, optionally data themselves, and also real-time Internet of Things (IoT) streaming data. The SODAS harvesting supports both data catalogs in standard formats like DCATv1/DCATv2/CKAN, and non-standard catalogs enabling data linkage and collection with various platforms. Third, in Analytics, users can search and execute analytics resources deployed in ADEs such as open API, Jupyter Notebook, workflow, and user-defined algorithms. This is related to the CKAN expansion to solve Problem 5, and we describe it in more detail in Section 4.5. Storage, a sub-menu of Analytics, supports storage volume allocation and monitoring for portal administrators. Portal administrators can allocate various data storage spaces to users through this Storage menu, and control the unnecessary waste of resources through usage monitoring. Fourth, the Service menu includes Visualization, Format Conversion, and De-identification as auxiliary functions that can easily identify and pre-process data to be used for data analysis. With Visualization, we can express datasets in various ways through Google chart [22], D3 API [23], and Chart Builder [24] packages. Finally, the Information menu supports notices for users, a question-and-answer board, and an organization board that can be used for each organization.

There are four function blocks that are highly related to Open Data Portal: data distribution, authentication/authorization, data storage management, and resource management/monitoring shown in Figure 5. Data distribution and resource management/monitoring blocks are new functions that are not available in the legacy CKAN, and data storage management and authentication/authorization blocks are extensions of the legacy CKAN. First, the data distribution block focuses on purchasing/selling processes such as shopping carts and billing/payment systems for data shopping. Next, the resource management/monitoring block includes resource allocation and resource usage monitoring functions for each organization/user for the platform administrator. With these functions, the administrator can control resources such as various computing infrastructures, datasets, and analysis algorithms.

In the legacy CKAN, an extension plug-in called Datastore [1] must be installed to store the data registered by users in the database or file system. However, this storing function is essential for open data platforms, and accordingly, we design the data storage management block to extend the data storage function based on Datastore, enabling the integrated management of various storages such as HDFS, NoSQL, and RDBMS. The authentication/authorization block is an extension of the CKAN's user registration, login, and user management, and we additionally implement SSO and user authentication based on Keycloak [25] to provide higher security.

4.3. Open Data Reference Model

The second component, Open Data Reference Model, consists of the governance elements which should be referred to for data management infrastructure and utilization of various kinds of datasets and interoperability among platforms. SODAS provides a separate portal for managing Open Data Reference Model based on DCATv2 support and extendable DataMap as the core strategies. This component includes domain classification systems and the extended metadata information for each category of the domain classification system. In addition, the defined classification system and various meta-information are used for data search, analysis, distribution, and information exchange among key features of SODAS. Figure 7 shows the main functions and the operation process of the Open Data Reference Model.

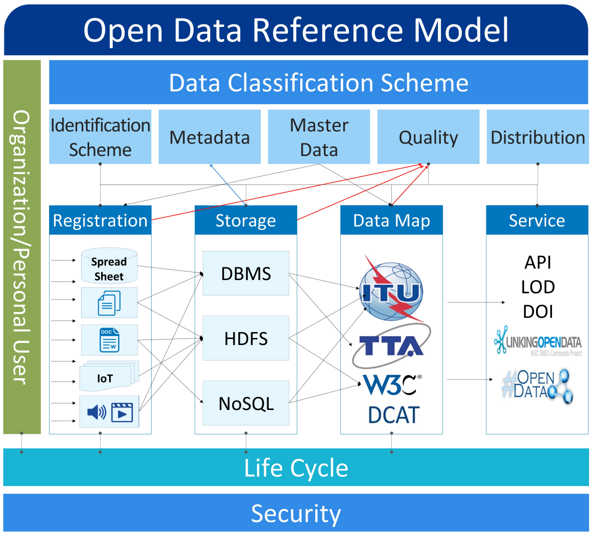


Figure 7. Main operations of Open Data Reference Model.

The main functions of the Open Data Reference Model are as follows. First, the data classification system supports catalog mapping and the classification system definition that can express hierarchical relationships. The classification system consists of a number of categories, and we can add or extend hierarchical categories by inheriting the metadata of the upper category. The classification system and catalog of SODAS basically follow DCATv2, and we also present a flexible form of extendable DataMap to define additional attributes. Through this structure, SODAS can map and link catalogs of more diverse domains than legacy CKAN. Second, metadata and master data mean the function of defining metadata based on the data classification system. In particular, master data refers to the essential metadata commonly used in business models among metadata. The defined meta-information is used for key functions of data distribution portals such as data registration, storage, and search/utilization. SODAS allows DCATv2-based metadata definitions so as to describe data in more detail. Third, the data quality function defines and manages quality indicators and evaluation methods according to the data characteristics. In Open Data Reference Model, we can define quality criteria/quality indicators/quality evaluation methods by classifying metadata and structured/unstructured data. We can also extend quality standards based on the data, supporting data standard format management and quality management tool interworking.

Function blocks that are highly related to Open Data Reference Model are quality/standard management and extendable DataMap of Figure 5, which are all newly designed functions by SODAS. First, the quality/standard management block standardizes various meta-information of data registered in the SODAS portal based on DCATv2, and manages quality standards and verification rules to provide high-quality data. Next, the extendable DataMap block includes a new data catalog that conforms to DCATv2 and Profile Vocabulary [26] standards to support various attribute definitions compare to existing platforms. The main functions based on extendable DataMap include metadata conversion tools and catalog sharing/distribution. In SODAS, we present a new catalog definition standard that improves the legacy CKAN through extendable DataMap. Additionally, the metadata conversion tool and catalog distribution functions can provide integration metadata defined differently and interwork with various open data platforms.

4.4. DataMap Publisher

DataMap Publisher creates metadata based on the data catalog managed by Open Data Reference Model and performs data harvesting based on the defined metadata. This corresponds to the Harvesting menu of the Open Data Portal. Generally, data platforms built by governments or private sectors often provide only their own data. Although we can use data harvesting through an extension plug-in like CKAN, it is rarely used due to functional errors or connection problems. In addition, CKAN and other open data platforms currently cannot share data due to different catalogs and interworking criteria. To address this problem, SODAS improves the legacy CKAN plug-in to enable DCAT-based data harvesting and supports a real-time data harvesting function. Figure 8 shows the harvesting operation of DataMap Publisher.

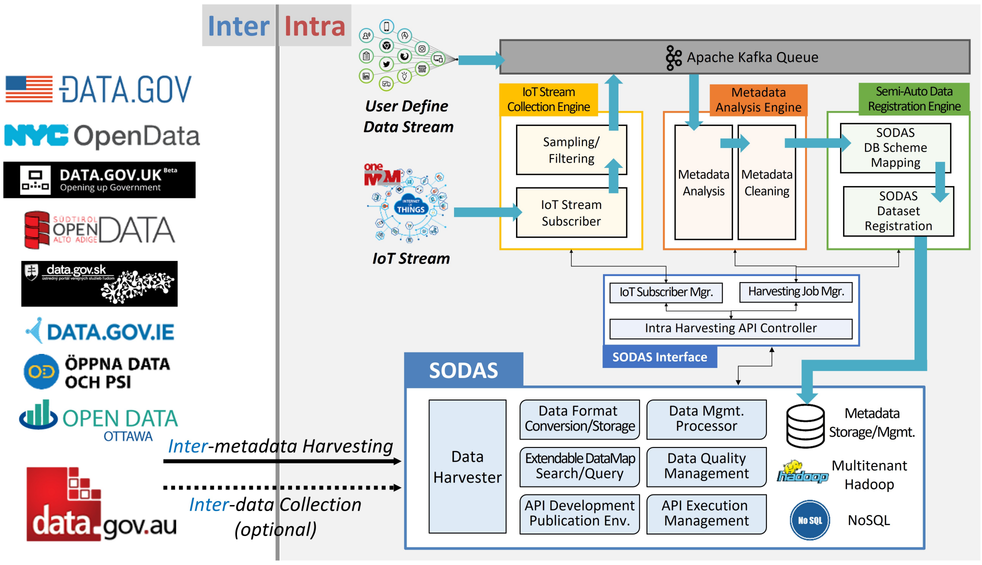


Figure 8. Harvesting operation of DataMap Publisher.

The Harvest from the legacy CKAN collects only metadata, hence, we cannot use actual data if there is a problem in the original site. We also found fatal errors in the Harvest plug-in, wherein the deleted data information still remains in the database and the harvest job could not be stopped [27]. DataMap Publisher includes Inter-harvester, which fixes errors in the existing plug-in and improves its functionality to collect data from the original site as well. We also designed a new Intra-harvester that supports real-time data collection such as large-capacity sensors and logs in various domains along with file data registered in the SODAS portal.

We also provide a metadata conversion tool as a new method of converting data catalogs based on an extendable DataMap for linking with other public or private data platforms. As shown in Figure 9, the metadata conversion tool maps and converts existing standard/non-standard catalogs to DCAT standards and distributes them as RDF files. Through this tool, the SODAS portal enables the harvesting of various open data platforms. Using the converted data catalogs, we can easily refer to open data of heterogeneous systems when searching for related data or building a new platform. As a result, the proposed metadata conversion tool enables stable interconnection between disparate systems as well as between CKANs.

The main function blocks of DataMap Publisher are data harvesting and extendable DataMap. Data harvesting is an improved and extended block of the CKAN's Harvest plug-in and includes all modules of Inter-harvester and Intra-harvester shown in Figure 8. Extendable DataMap consists of the metadata conversion tool and catalog sharing/distribution functions.

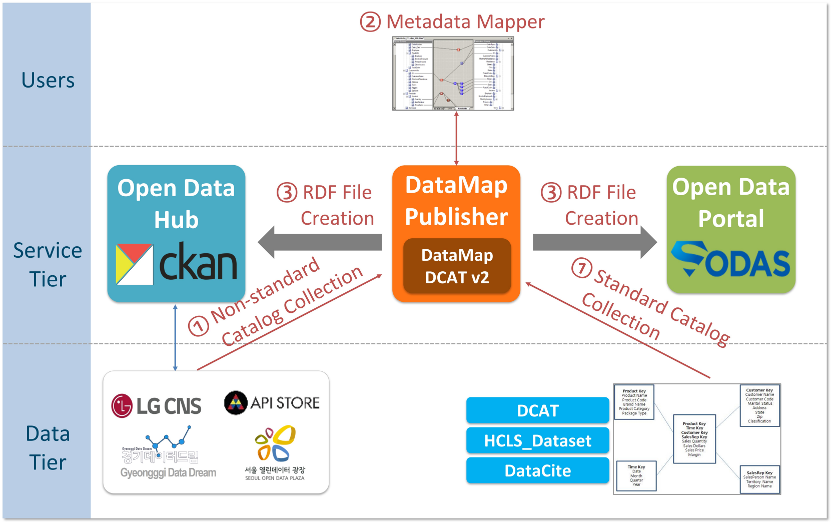


Figure 9. Operation of metadata conversion tool.

4.5. ADE Provisioning

Low utilization (Problem 5) is the major reason for lowering the utilization of open data as well as open data platforms. To solve this problem, we present ADE Provisioning, a new integrated analytics and development environment that allows users to easily analyze and utilize their own data on top of the SODAS platform. Figure 10 shows an operational diagram of ADE Provisioning. As shown in the figure, SODAS enables users to perform user-defined algorithm-based analysis and TensorFlow-based machine learning/deep learning analysis using Eclipse Theia [28] and Jupyter Notebook [29]. In addition, API registration and distribution functions are also supported through Swagger [30], which can manage OpenAPI specification standards. Similar to Open Data Reference Model, ADE Provisioning is provided as a separate Web service from Open Data Portal. The results developed and distributed here can be checked in the Category > Data Service menu of Open Data Portal, and users can purchase and execute the algorithms directly from the Analytics menu.

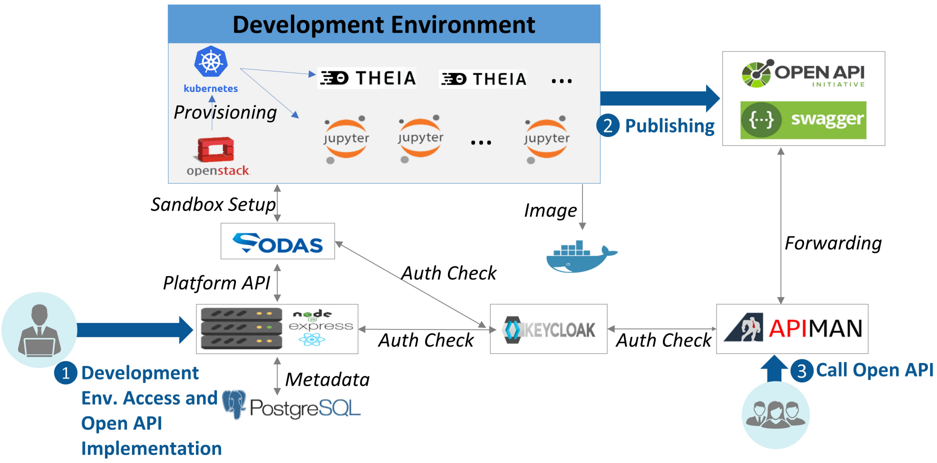


Figure 10. Operational structure of ADE Provisioning.

The main operation process of ADE Provisioning is as follows. First, when a user logs into the ADE Provisioning portal, a sandbox is allocated based on Openstack [31] and Kubernetes [32] so that the user can run the development environment on SODAS through authentication. Next, the user directly develops the required algorithm by running Theia or Notebook environment in the assigned virtual sandbox environment. When the development is completed, the user can create and distribute API documents according to the OpenAPI specification standard. Finally, the published algorithms can be checked on Open Data Portal, and other users can purchase and run the published algorithms on the portal to analyze their own data.

The cloud development environment and service provision blocks of ADE Provisioning are new features proposed by SODAS, and we designed both blocks based on the latest open-source frameworks that are actively used. The first block, the cloud development environment, supports user-specific sandbox analysis and development environments through APIs, Notebook, and workflow development tools. It also includes supplementary functions such as Openstack and Kubernetes-based sandbox configuration and resource allocation/management for stable development environment operation. The second block, service provision, provides publication and registration of algorithms and services developed through ADE Provisioning. Using this block, we can register the developed algorithms in the Category and Service menu of the Open Data Portal, and provide more accurate search algorithms and visualizations of the relationship between search results.

5. System Implementation and Substantiation

5.1. Results of SODAS Platform Development

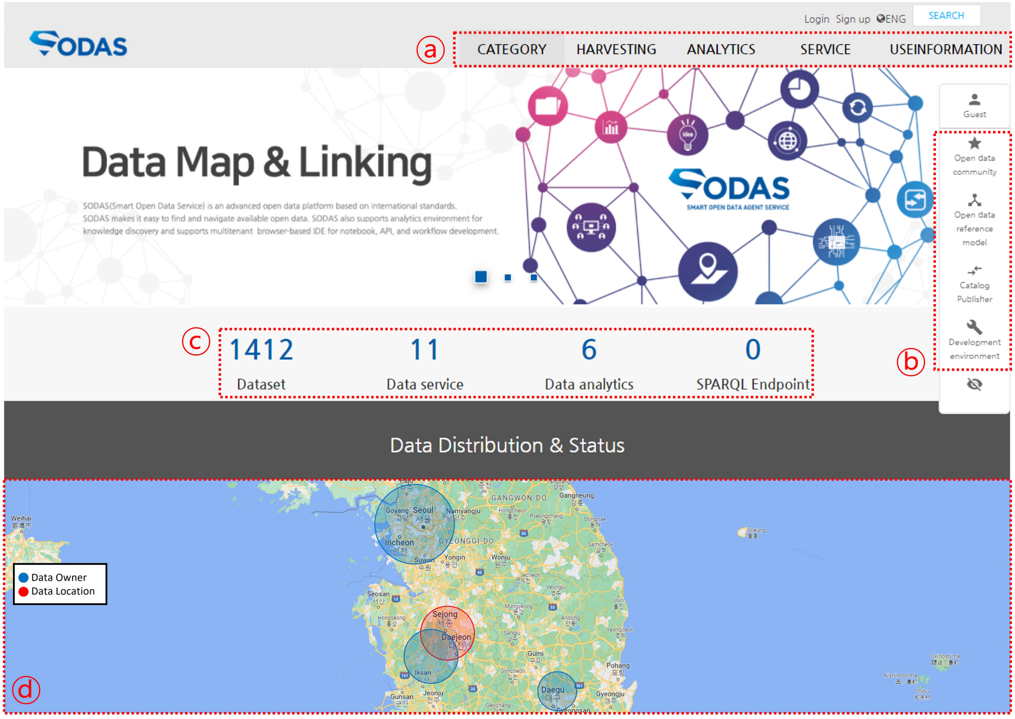
In this section, we show the actual results of implementing the four components of SODAS. Because we derive SODAS components from different goals and functions, the implementation environment is very complex. Table 4 summarizes the core open-source framework used for the component implementation. SODAS integrates each function implemented with these open sources into Node.js, and we can see the result through the actual Web UI. We release some of the development results as CKAN extension (Resource Authorizer (resourceauthorizer), Keycloak Authenticator (keycloak), HDFS Storage Manager (hdfs), TripleStore Storage Manager (jena) accessed on 26 January 2023 (http://extensions.ckan.org/extension/[short-name]/)). Furthermore, readers can see the actual demonstration video of SODAS on our github (https://github.com/sodas-etri/demo/tree/master/videos, accessed on 26 January 2023). In this section, we validate the effectiveness of SODAS through the implementation results of each component and its actual applications.

Table 4. Implementation environments of SODAS.

Component	Open-Source (License)	Purpose of Usage
Open Data Portal	Keycloak (Apache)	Integrated user authentication
	Loopback (MIT), Play (Apache)	System REST API provisioning
	Hadoop (Apache), Hbase (Apache), Jena (Apache), MongoDB (AGPL), PostgreSQL (PostgreSQL)	Data storage by data size/purpose
	Spark (Apache)	Data analysis
	Grafana (Apache), InfluxDB (MIT)	Platform monitoring UI and log storage
Open Data Reference Model	Express (MIT), Axios (MIT)	System API interworking and provision
	Ejs (Apache), Multer (MIT)	Web UI and file upload development
DataMap Publisher	Kafka (Apache), Jena (Apache)	Data stream queuing, RDF storage
	Spring (Apache)	System API development
ADE Provisioning	Express (MIT)	System REST API provisioning
	Notebook (Jupyter), Eclipse Theia (Apache)	Cloud-based user analysis/development
	Openstack (Apache), Kubernetes (Apache), Docker (Apache)	Virtual machine/execution environment management

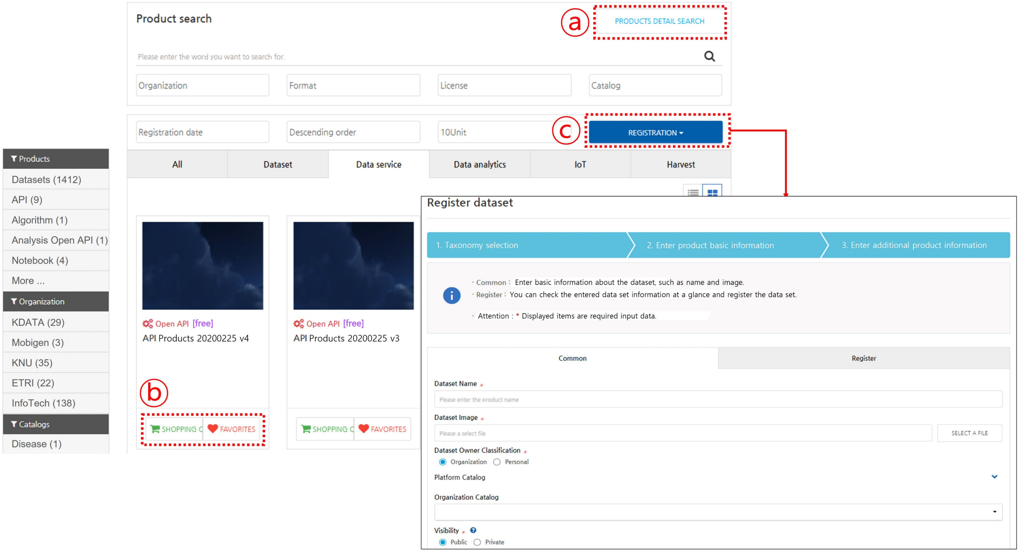
Open Data Portal is the main service of SODAS, and it connects with Open Data Reference Model, DataMap Publisher, and ADE Provisioning through the link menu. Figure 11 shows the main screen of the SODAS data portal. As shown in the figure, the main screen is composed of ⓐ data/service resources search and board menu, ⓑ components connection link, ⓒ resource statistics information, and ⓓ map-based resource metadata visualization.

First, using the Category menu, the users can search for datasets, data services, data analysis (user-defined algorithms), and harvest resources as shown in Figure 12. SODAS provides a more accurate advanced search of ⓐ for experts, and users can add each resource to the shopping cart or favorites in ⓑ. Users can register their own datasets and APIs through ⓒ. On the registration screen, they can input various metadata such as data name, language, keyword, price, and catalog information required for the data portal.



SODAS(Smart Open Data Service) is an advanced open data platform based on international standards. SODAS makes it easy to find and navigate available open data. SODAS also supports analytics environment for knowledge discovery and supports multitenant browser-based IDE for notebook, API, and workflow development.

Figure 11. Main screen of SODAS data portal.



- Common : Enter basic information about the dataset, such as name and image.
- Register : You can check the entered data set information at a glance and register the data set.

- Attention : * Displayed items are required input data.

Dataset Name *
Please enter the product name

Dataset Image *
Please a select file

Dataset Owner Classification *
● Organization ○ Personal

Platform Catalog

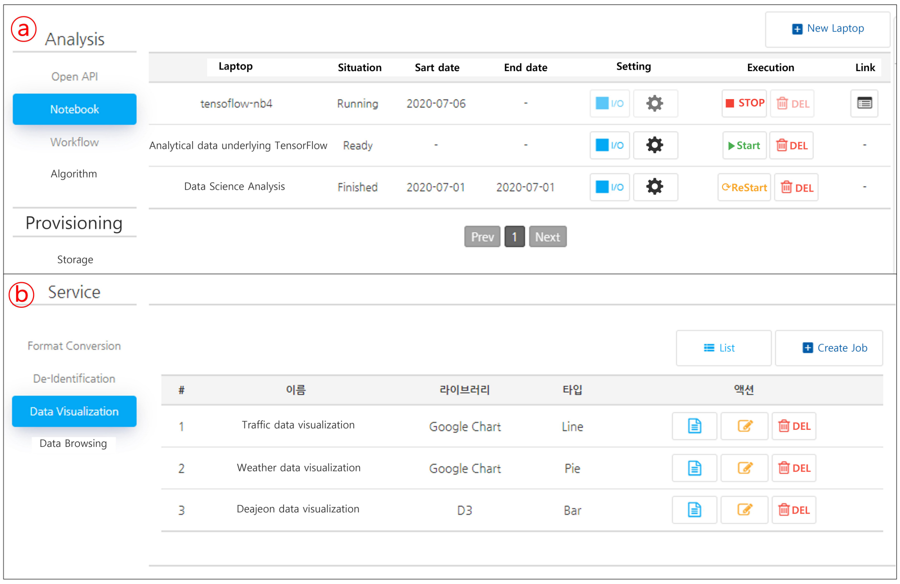
Organization Catalog

Visibility *
● Public ○ Private

Figure 12. Functions of the Category menu.

Next, in the Harvesting menu, users can check the data connection status with other platforms according to various criteria, and search for appropriate harvesting. Currently, SODAS supports interworking with DCATv1, DCATv2, SODAS, CKAN, and all data platforms that use non-standard catalogs.

Finally, using the Analytics and Service menus, users can execute various analyses, data conversion, purification, and visualization using user-defined algorithms published in ADE Provisioning. Figure 13 shows example screens of Analytics and Service menus. Using the Analytics menu of ⓐ, and we can confirm the settings of input/output/virtual environment and execution management of analysis algorithms developed with Open API and Notebook. Using the Data Visualization menu of ⓑ, a sub-menu of Service, users can directly visualize their own data through open-source visualization libraries such as Google Chart and D3.js, and easily identify the characteristics of each data.



Laptop	Situation	Sart date	End date	Setting	Execution	Link
tensoflow-nb4	Running	2020-07-06	-	I/O ⚙	■ STOP DEL	▤
Analytical data underlying TensorFlow	Ready	-	-	I/O ⚙	▶Start DEL	-
Data Science Analysis	Finished	2020-07-01	2020-07-01	I/O ⚙	⟳ReStart DEL	-

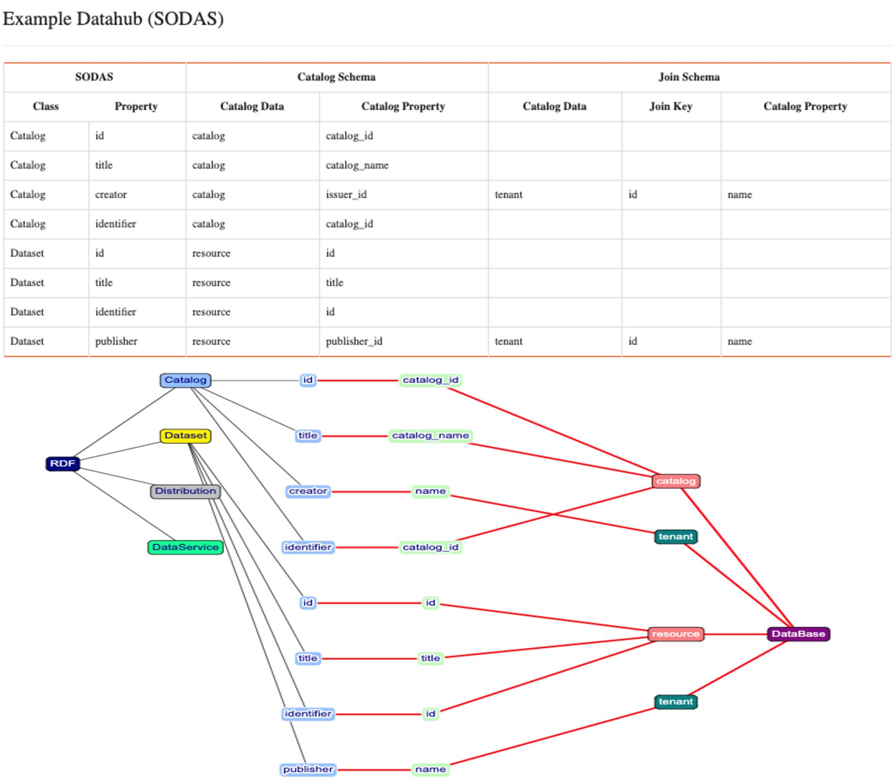
#	이름	라이브러리	타입	액션
1	Traffic data visualization	Google Chart	Line	DEL
2	Weather data visualization	Google Chart	Pie	DEL
3	Deajeon data visualization	D3	Bar	DEL

Figure 13. Example screens of the Analytics and Service menus.

Open Data Reference Model supports the creation and management of classification systems, catalogs, and vocabulary based on extendable DataMap. This component also includes quality indicators, quality items, and business rule management for improving data quality. Using the Classification system management menu, we can create and manage extendable DataMap, one of SODAS' core strategies. We can also check the metadata and master data of the item and add/modify/delete the necessary metadata properties on this menu.

DataMap Publisher provides catalog mapping functions between open data platforms using the metadata standard created by Open Data Reference Model. SODAS harvests data from other platforms based on these catalogs. Figure 14 shows an example screen of mapping the SODAS standard and other platform catalogs using DataMap Publisher. As shown in the figure, users can designate the catalog scheme to be mapped based on the DCAT class and attribute in the Catalog management menu, and also define the join relationship of metadata. In this menu, we can easily know the mapping information through an intuitive graph that visualizes the connectivity among classes and properties.

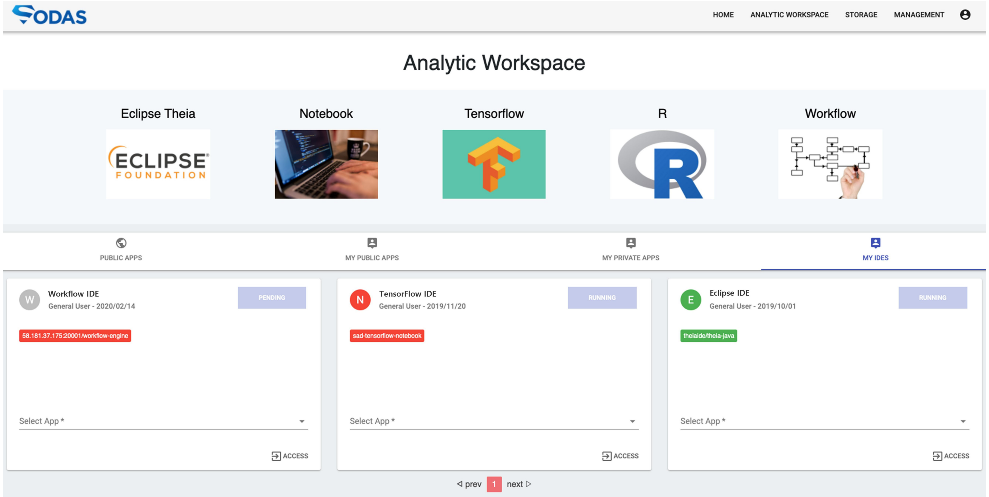
For easy access to SODAS, we implement ADE Provisioning as a cloud-based Web UI. Figure 15 shows an example screen of the Analysis environment menu, a representative menu of ADE Provisioning. It currently provides a user-defined analysis through Theia, Notebook, TensorFlow, R, and Workflow. Users can check the list of their own data in this menu and develop analysis algorithms using the online development environment.



Example Datahub (SODAS)

SODAS		Catalog Schema		Join Schema		
Class	Property	Catalog Data	Catalog Property	Catalog Data	Join Key	Catalog Property
Catalog	id	catalog	catalog_id			
Catalog	title	catalog	catalog_name			
Catalog	creator	catalog	issuer_id	tenant	id	name
Catalog	identifier	catalog	catalog_id			
Dataset	id	resource	id			
Dataset	title	resource	title			
Dataset	identifier	resource	id			
Dataset	publisher	resource	publisher_id	tenant	id	name

Figure 14. Example screen of catalog mapping using DataMap Publisher.



W Workflow IDE
General User - 2020/02/14
PENDING
58.181.37.175:20001/workflow-engine
Select App *
ACCESS

N TensorFlow IDE
General User - 2019/11/20
RUNNING
sad-tensorflow-notebook
Select App *
ACCESS

E Eclipse IDE
General User - 2019/10/01
RUNNING
theiaide/theia-java
Select App *
ACCESS

◁ prev 1 next ▷

Figure 15. Example screen of ADE Provisioning main menu.

To evaluate the effectiveness of SODAS, we have experimented with two measures: dataset collection efficiency and related dataset detection rate as follows. We implemented the experiments using Nodejs 8.12.0 and Python 2.7.12 on a single server with Intel Xeon Silver 4110 CPU@2.10 GHz, 125GB RAM.

- Dataset collection efficiency: We measure the number of datasets finally gathered by liking SODAS with other open data platforms to verify the efficiency of collecting datasets from SODAS. As a result of connecting with 7 Korean and 16 other countries' data distribution portals through DataMap Publisher, a total of 261,683 datasets are collected. This result means that SODAS can more easily collect open data by interconnecting with the existing CKAN-based platform. In addition, since SODAS directly stores data with an improved harvester, it can provide data reliably even if the original data link is damaged.

- Related dataset detection rate: This detection rate is to evaluate how well SODAS finds datasets similar to those selected by the user through Open Data Reference Model. For evaluation, we prepare correct answer sets for each field and perform experiments to find datasets related to the dataset given as a query. In this experiment, the measured value represents the percentage of the search results including correct answers. According to the experimental results, SODAS shows an average of 74.65% of the related dataset detection rate. SODAS utilizes these results to support users with high-accuracy data recommendation functions without unnecessary search processes.

Summarizing the experimental results, we can see that SODAS ensures interoperability with the existing platforms through extended standard support, thereby efficiently providing data collection and management, which are the core functions of the open data platform. SODAS also increases platform user convenience by improving the detection performance of related data based on systematic metadata management and reference models.

5.2. SODAS Application Sites

Several organizations and companies have actually used SODAS to provide their own data distribution and management services. Figure 16 shows four real sites using SODAS for the actual service. Figure 16ⓐ is a Data Store operated by KDATA, and currently open data platforms in more than 16 countries are linked, supporting transactions of files, APIs, and image data. Figure 16ⓑ shows a financial data exchange market of the Financial Security Institute and it applied SODAS to manage the sales/purchase of data in the financial field. Figure 16ⓒ, PartnerHub, is the environmental open data hub of the PARTNER project for medical data. Figure 16ⓓ shows the cultural big data platform of Korea Culture Information Service Agency. These organizations have built their own sites based on SODAS to collect, register, and sell big data in the relevant field, and are actively promoting and operating concurrently to promote data sharing in their specific fields.

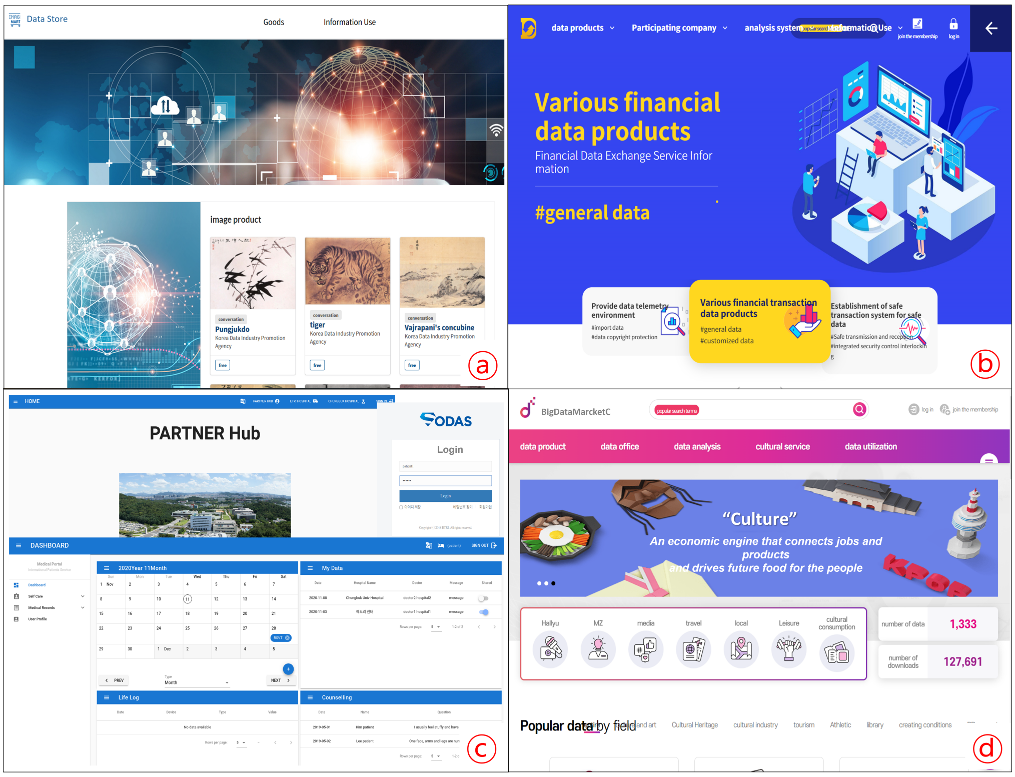


Figure 16. SODAS-based actual operation services.

To show how the application site utilizes SODAS, we describe PartnerHub in more detail. PartnerHub is a data analysis portal in the medical field based on SODAS as shown in Figure 17. The main users of PartnerHub are doctors, medical scientists, and patients, and the platform provides data construction, analysis, and decision support functions based on the patient's medical information. Since data security is very important in the medical field, the platform strengthens user authentication by combining Single Sign-On (SSO) with SODAS. PartnerHub is one of the representative examples of applying SODAS to the private environment.

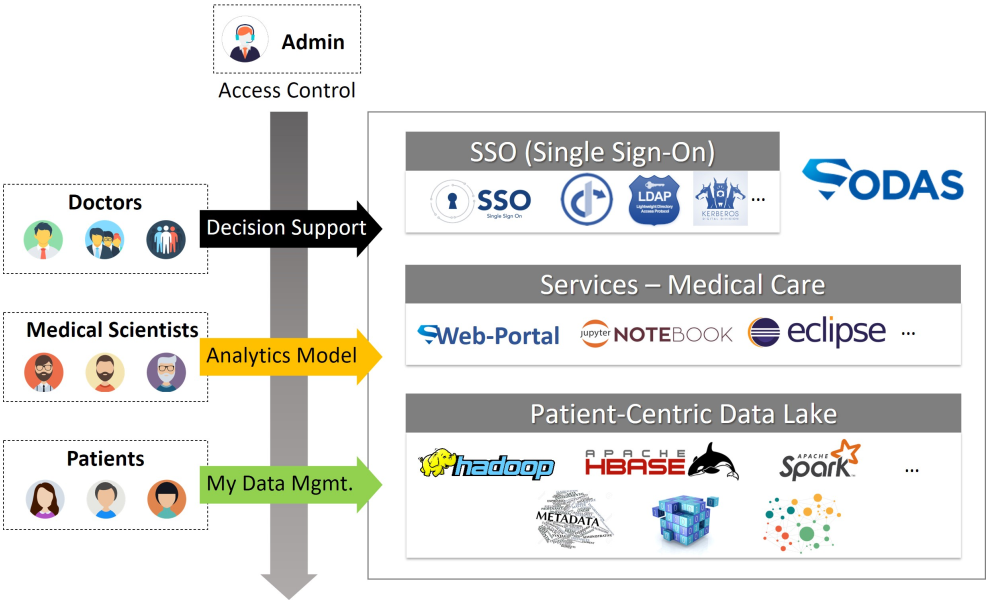


Figure 17. Working scenarios of PartnerHub applications.

6. Conclusions

In this study, we designed and implemented SODAS, a novel open data platform, based on CKAN and DCATv2 to share open data more actively and easily. First, after our extensive analysis of the legacy CKAN, we presented five problems, and then we introduced three core strategies to solve these problems. Second, based on the proposed core strategies, we newly defined SODAS as four components. Third, we further defined nine function blocks in detail to actually develop the four components and implemented the blocks as open-source to confirm the functions of SODAS. Fourth, we applied and operated SODAS to data distribution services of several actual domains and proved its practical effectiveness as a data hub. We believe that SODAS is the first attempt to activate the data hub and improve data utilization without domain restrictions based on the CKAN expansion and extendable DataMap. SODAS is also an all-in-one platform for storing, managing, analyzing, and distributing real-time IoT data generated at high-speed and large volumes, and can contribute to data acquisition and release in various fields such as healthcare, energy, and smart cities. In the future, we plan to clearly derive the general problems of open data platforms and present an improved version of SODAS as a solution for these problems. We will also study the application of high-performance communication technologies, such as Remote Direct Memory Access (RDMA), to improve the overall performance of SODAS, as it is a modular structure. Afterward, we will evaluate the user-side execution performance of the improved SODAS and prove that SODAS can be effectively utilized in real services.

Author Contributions: Author Conceptualization, formal analysis, and software, H.W. and J.H.; writing—original draft preparation and editing, M.-S.G.; writing—review, editing and validation, Y.-S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korean government(MSIT) (No. 2020-0-00077, Core Technology Development for Intelligently Searching and Utilizing Big Data based on DataMap) and the Korea Institute for Advancement of Technology(KIAT) grant funded by the Korea government(MOTIE) (No. N0002452, PARTNER: Patient-care Advancement with Responsive Technologies aNd Engagement togetheR).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. CKAN Documentation. Available online: http://docs.ckan.org/ (accessed on 26 January 2023).
2. Open Government Platform (OGPL). Available online: https://ogpl.github.io/ (accessed on 26 January 2023).
3. Socrata. Available online: https://dev.socrata.com/ (accessed on 26 January 2023).
4. Albertoni, R.; Browning, D.; Cox, S.; Beltran, A.G.; Perego, A.; Winstanley, P. *Data Catalog Vocabulary (DCAT)—Version 2*; The World Wide Web Consortium (W3C): Cambridge, MA, USA, 2020.
5. PARTNER Project. Available online: https://itea3.org/project/partner.html (accessed on 26 January 2023).
6. Open Knowledge Foundation (OKFN), Why Open Data. Available online: https://okfn.org/opendata/why-open-data/ (accessed on 26 January 2023).
7. Auer, S.; Bizer, C.; Kobilarov, G.; Lehmann, J.; Cyganiak, R.; Ives, Z. DBpedia: A Nucleus for a Web of Open Data. In Proceedings of the International Semantic Web Conference, Busan, Korea, 11–15 November 2007; pp. 722–735.
8. Li, Y.; Jiang, Y.; Goldstein, J.C.; Mcgibbney, L.J.; Yang, C. A Query Understanding Framework for Earth Data Discovery. *Appl. Sci.* **2020**, *10*, 1127. [CrossRef]
9. Yang, P.; Evans, J.; Cole, M.; Marley, S.; Alameh, N.; Bambacus, M. The Emerging Concepts and Appli-cations of the Spatial Web Portal. *Photogramm. Eng. Remote Sens.* **2007**, *73*, 691–698. [CrossRef]
10. Vítor, G.; Rito, P.; Sargento, S. Smart City Data Platform for Real-time Processing and Data Sharing. In Proceedings of the IEEE Symp. on Computers and Communications, Athens, Greece, 5–8 September 2021; pp. 1–7.
11. Li, C.; Zhang, J.; Kale, A.; Que, X.; Salati, S.; Ma, X. Toward Trust-Based Recommender Systems for Open Data: A Literature Review. *Information* **2022**, *13*, 334. [CrossRef]
12. Moreno, A.; Molano-Pulido, J.; Gomez-Morantes, J.E.; Gonzalez, R.A. ADACOP: A Big Data Platform for Open Government Data. In Proceedings of the International Conference on Theory and Practice of Electronic Governance, Guimarães, Portugal, 4–7 October 2022; pp. 369–375.
13. Janssen, M.; Charalabidis, Y.; Zuiderwijk, A. Benefits, Adoption Barriers and Myths of Open Data and Open Government. *Inf. Syst. Manag.* **2012**, *29*, 258–268. [CrossRef]
14. Winn, J. Open Data and the Academy: An Evaluation of CKAN for Research Data Management. In Proceedings of the International Association for Social Science Information Services & Technology, Cologne, Germany, 28–31 May 2013; pp. 28–31.
15. Junar. Available online: https://www.junar.com/ (accessed on 26 January 2023).
16. Momjian, B. *PostgreSQL: Introduction and Concepts*; Addison-Wesley: New York, NY, USA, 2001; Volume 192.
17. Copeland, R. *Essential SQLAlchemy*; O'Reilly Media: Cambridge, MA, USA, 2008.
18. O'Neil, E. Object/Relational Mapping 2008: Hibernate and the Entity Data Model (EDM). In Proceedings of the International Conference on Management of Data, ACM SIGMOD, Vancouver, BC, Canada, 10–12 June 2008; pp. 1351–1356.
19. Smiley, D.; Pugh, E.; Parisa, K.; Mitchell, M. *Apache Solr Enterprise Search Server*; Packt Publishing Ltd.: Birmingham, UK, 2015.
20. Guha, R.V.; Brickley, D.; Macbeth, S. Schema.org: Evolution of Structured Data on The Web. *Commun. ACM* **2016**, *59*, 44–51. [CrossRef]
21. Brickley, D.; Burgess, M.; Noy, N. Google Dataset Search: Building a Search Engine for Datasets in An Open Web Ecosystem. In Proceedings of the International World Wide Web Conference Committee, San Francisco, CA, USA, 13–17 May 2019; pp. 1365–1375.
22. Zhu, Y. Introducing Google Chart Tools and Google Maps API in Data Visualization Courses. *IEEE Comput. Graph. Appl.* **2012**, *32*, 6–9. [PubMed]
23. D3.js. Available online: https://d3js.org/ (accessed on 26 January 2023).
24. Chart Builder. Available online: https://github.com/plotly/react-chart-editor (accessed on 1 March 2023).
25. Keycloak. Available online: https://www.keycloak.org/ (accessed on 26 January 2023).
26. Profiles Vocabulary. Available online: https://www.w3.org/TR/dx-prof/ (accessed on 26 January 2023).
27. Kim, D.; Gil, M.-S.; Nguyen, M.C.; Won, H.; Moon, Y.-S. Comprehensive Knowledge Archive Network Harvester Improvement for Efficient Open-Data Collection and Management. *ETRI J.* **2021**, *43*, 835–855. [CrossRef]
28. Eclipse Theia. Available online: https://theia-ide.org/ (accessed on 26 January 2023).
29. Jupyter Notebook. Available online: https://jupyter.org/ (accessed on 26 January 2023).
30. Swagger. Available online: https://swagger.io/ (accessed on 26 January 2023).

31. Sefraoui, O.; Aissaoui, M.; Eleuldj, M. OpenStack: Toward an Open-Source Solution for Cloud Computing. *Int. J. Comput. Appl.* **2012**, *55*, 38–42. [CrossRef]
32. Burns, B.; Grant, B.; Oppenheimer, D.; Brewer, E.; Wilkes, J. Borg, Omega, and Kubernetes. *Queue* **2016**, *14*, 70–93. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.