Big Data-as-a-Service: Definition and architecture

Xinhua E [1], Jing Han [2], Yasong Wang [2], Lianru Liu [2]

[1]School of Computer, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]School of Information and Communication engineering,
Beijing University of Posts and Telecommunications, Beijing 100876, China
exinhua@foxmail.com, {Hanj12, Liulr18, wangys274}@chinaunicom.cn

Abstract: Big Data-as-a-Service (BDaaS) is a core direction in the age of big data to help companies gain intrinsic value from big data and innovative their business strategies. Based on analyzing technological challenges that BDaaS faces, firstly, a clear definition of BDaaS was given. After that a User Experience-oriented BDaaS Architecture was constructed .In addition, service processes of processing, analysis and visualization requests were described in detail. Contrast with conventional data services architectures, UE-BDaaSA supports for unstructured data, and provides a wide variety of service, such as analysis and visualization services, and it can better meet the needs of big data era.

Keywords: Big Data-as-a-Service; data services; definition; architecture; unstructured data

1 Introduction

With the rise of mobile internet technology, social network, and the worldwide rapid growth of data volume, Big Data era has arrived. Nowadays, big data has become an important trend of modern information technology, and sharing and analysis of big data would not only bring immeasurable economic value, but also play a significant role in promoting the development of society. Therefore, big data has been referred to as a new category of economic asset [1].

Existing research on big data were chiefly focus on key technologies, such as data storage, data processing, data analysis, and data visualization. However, this way has many disadvantages, including high cost, higher technical threshold, and profits are hard to be guaranteed. Therefore, along with the rapid development of service economy, a large number of third-party service providers have sprung up to provide consumers with big data storage, processing, analysis, visualization services which are dynamic, on-demand, and automated, with which users can get value from big data and only need to spend a small service cost.

So far the research of BDaaS is still in its infancy, it still faces challenges as follows: 1) Without a clear definition; 2) Lack of a clear classification; 3) There is no standardized, user experience based BDaaS architecture which can shield the complexity of data sources and operations. In order to solve these problems, the definition, classification and architecture of BDaaS would be in-depth studied.

Firstly, in Section II, the background of BDaaS was described, and the definition of BDaaS was given. Then the architecture of BDaaS platform was proposed in Section III. In Section IV, the classification and processing flow of data requests were discussed. In Section V, BDAAS architecture and traditional data services architecture were compared. Section VI concludes.

2 Related works

2.1 Big Data-as-a-service and Data services

December 2012, the world's authoritative market research institution Technavio released "Global Big Data-as-a-service Market 2012-2016"[2], it reported that companies need BDaaS solution to automatically track their IT systems performance and behavior , also should use big data analysis to innovative business strategies, and to improve overall operational efficiency. At present, the companies who have occupied BDaaS market space are EMC, IBM, Microsoft, Amazon, Google, Snaplogic, Oracle, SAP, etc., which mainly provide big data storage and analysis services. For example, EMC offer enterprise for big data storage and analysis services. Greenplum [3] is data storage and analysis tool set of EMC, which consists of three parts: Greenplum Database, Greenplum HD and Isilon [4]. Greenplum database manages, storages and analyzes PB-level data. Greenplum HD is the commercial branch of Hadoop, it allows user to use Hadoop for Big Data Analytics without considering the complexity of Hadoop versions. Isilon clustered storage is a scale-out Network Attached Storage (NAS) platform, it can support storing 15PB data in a single file system and easy to manage; Amazon provides independent big data analysis services though AWS Marketplace[5]; Microsoft is also provide big data analysis service through Windows Azure MarketPlace [6]; Google offers Google BigQuery [7]to support big data analysis; SnapLogic [8] provides enterprises for big data processing service solutions which help them to obtain value by analyzing both business data and external data.

Before BDaaS produce, academia and industry has conducted a lot of research work in Data Services (DS). Data service is combination of Web services and data management technology. Different with Web services, data services encapsulate various heterogeneous data sources and descriptions in a uniform way to achieve

978-1-4799-0077-0/13/$31.00 ©2013 IEEE

cross-domain data integration.

However, there is a big difference between BDaaS and DS: 1) In aspect of data object, DS only supports structured data, while BDaaS support not only structured data, but also supports unstructured data based on building a unified unstructured data model; 2) In the respect of data object, DS follows the traditional Web service description method(such as WSDL), and only describe the service interface specification, while the service model of BDaaS(maybe OWL-S based) can cover the data characteristics of big data, such as data model, data quality, privacy etc.; 3).In the respect of service content, DS mainly for the "Data Providing Services", which means that it provides the raw data to the service consumer, to provide users with read-only BDaaS services, including data processing, retrieval, analysis and visualization, etc. .

2.2 Architectures

In the respect of architecture, [9] proposed an abstract data service architecture, which consists of five parts: Data Stores, External Model, Consuming Methods, Data/Metadata Requests and Data&Metadata. In addition, the principles mapping query requests submitted by service consumers to query statements of data storage system have been also be defined when building the model.

Some of the more famous architecture includes Microsoft WCF DS and Oracle's OSDI.

Windows Communication Foundation Data Services (WCF DS) [10] (formerly ADO.NET Data Services Framework) is Microsoft's Data Services Framework. WCF DS framework consists of three parts: Data Service Providers, Data Services Runtime and Hosting / HTTP listener. Moreover, WCF DS can support multiple types of data sources, such as relational database which is mapped to EDM model, the common language runtime (CLR) classes, or other OData data source being bound. Users can through standard HTTP verbs such as GET, PUT, POST and DELETE, to access or update data, after processing, WCF DS returns the results in JSON or ATOM format.

Oracle Data Services Integrator (ODSI) is Oracle's Data Services Platform [11] which can integrate multiple data sources, the modules of OSDI including Data sources, Data source API, Data Processing Engine, Caching, Security, Client API, Data Service Development Tools, and Administration Console.

It can be seen that these two commercial data services architecture is positioned to provide data owner a platform manipulate (maybe get, update, or delete) data in real time. However, the goal of BDaaS architecture is to offer big data service providers an abstract reference model to guide building big data service system, in which the service consumers would get result(maybe analytical results or infographic) only if they inputted their requests.

In order to construct big data service architecture, the following issues must be addressed:

In the respect of module designing: the abstract architecture proposed in paper [9] great significance to design BDaaS architecture. In view of unique characteristics big data have such as massive, stored in different administrative domains and wide variety of data sources, in order to provide data processing, retrieval, analysis and visualization services, which types of data BDaaS should support, how to design modules, all this issue should be examined.

Big data contains not only a large amount of structured data, but more unstructured data. The data model should be redesigned to enable BDaaS architecture support the unstructured data.

The user experience is a key indicator of service capabilities, and user behavior monitoring is the premise to improve the user experience. So a set of rules and processes have to be defined and orchestrated to monitor and analyze the user behavior.

3 Definition and Architecture

3.1 Definition of BDaaS

BDaaS is one of the new research direction in the field of big data, a Clear definition have important implications for the following research. Therefore, this paper defines BDaaS as follows:

Difinition1 Generalized definition of BDaaS： Big data-as-a-Service (BDaaS) is a new way to get insight from big data, it is also a new form of service economy. By encapsulating heterogeneous data as a service, it shields the differences on data structure and definitions, and the users just concern on what they want and get the service whenever and wherever to store, search, analyze and visualize the data.

Difinition2 Narrow definition of BDaaS：A BDaaS is a functional unit with a clear contract and independent function which can be deployed independently. A BDaaS can be represented by a three-tuple *<ID, Prof, Endp>*，*ID* uniquely identifies a BDaaS; *Prof* describes the functions of the service, such as service providers, service contract, privacy items, Qos, etc.; *Endp* is the endpoint set of BDaaS through it to interact to outside, each endpoint *Endpi* can be represented by a seven-tuple *<EIDi, Souri, Funci, Extni, Parai, Transi, Condi>*,

EIDi denotes the identity of the *i*-th endpoint;

Souri denotes the data sources set of the *i*-th endpoint;

Funci is the function set of the *i*-th endpoint, through which can get data from outside, and the opration is represented by *f*;

Extni is the function set of the *i*-th endpoint, through which can output result to outside, and the opration is represented by *e*;

Parai is the parameter set of the *i*-th endpoint, the parameter relative to Funci is called input parameters

which can be represented by *Parai(f)*，and the parameter relative to Extni is called output parameters which can be represented by *Parai(e)*;

Transi is the data operations set of the *i*-th endpoint;

Condi is the behavior restriction of the *i*-th endpoint, Condi is represented by a four-tuple *<init, preCond, postCond, effect>, init, preCond, postCond*, effect respectively represent the initial condition, pre-condition, post-condition and the event would be triggered if the service was evaluated successfully.

3.2 User Experience-oriented BDaaS Architecture

Difinition3 User Experience-oriented BDaaS Architecture (UE-BDaaSA): UE-BDaaSA is a logical architecture which provides services such as processing, analysis, retrieval and visualization services by shielding heterogeneous data sources and data operation differences, and composes various BDaaS dynamically according to the application requests and user behavior.

The user of UE-BDaaSA means all outside entities that would interact to the architecture, it contains service provider, service consumer, and data owner in which the service consumer can be divided into normal user, professional user and applications. In practice, the data owner and service provider may be the same entity some occasion.

In the UE-BDaaSA, the user entities are modeled as follows:

Service Provider (SP): SP is represented by a three-tuple, SP = *<ID,RSL,IL>*, *ID* uniquely identifies a user, *RSL* refers to a list of registered services, *IL* refers to the service provider's interest area of.

Service Consumer (SC): SC is represented by a four-tuple, SC = *<ID,ASL,IL,RL>*, *ID* uniquely identifies a user, *ASL* refers to authorized services list, *IL* expressed areas of interest, *RL* represents the task list which contain the recent requests of service consumer.

Data Provider (DP): DP is represented by a four-tuple, DP = *<ID,RSL,DSL,IL>*, *ID* uniquely identifies a user, *RSL* refers to a list of registered services, *DSL* means that the data provider's data source list, *IL* refers to its area of interest.

UE-BDaaSA is a high-level conceptual architecture, which consists of three layers, namely, data storage layer, service engine layer and application layer. With this architecture, on the one hand, data owners can ensure that the data will be accessed appropriate and limited, and can predict the impact these data accesses brought to its infrastructure; on the other hand, data consumers and application developers can also gain a way to access data and mechanism to help themselves to integrate data from multiple data sources. The architecture is shown in Figure 1, and each layer was described as follows:

Data Service Store (DSS): DSS consists of a storage system and a data model module. Storage system store relational data, non-relational database data and unstructured data, and data model module extracts the data model for each type of data source, especially for unstructured data, it should be structuralize.

Data Services Engine (DSE): DSE is the core component of the UE, its functions are as follows: data services registration, user entity and behavioral modeling, BDaaS modeling, BDaaS dynamically generated, BDaaS composition, request distribution, application requests decomposition, request results assembly. When creating a BDaaS, users can define external schema mapping rules and descriptions, DSE provides function to generate model, which will automatically map the external model to data object of data source.

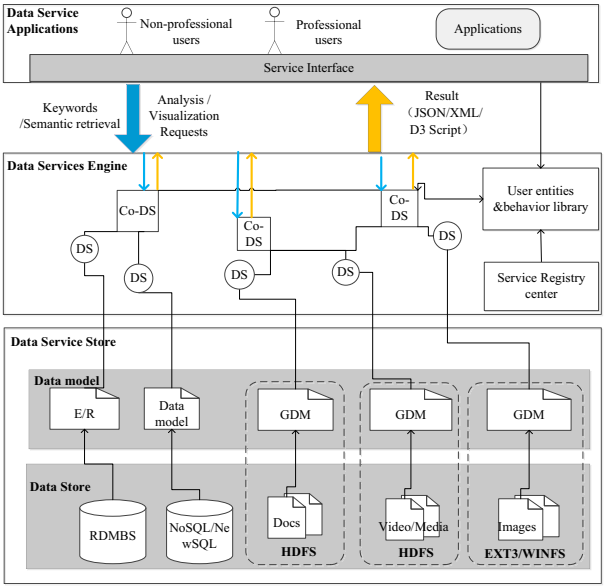

Figure 1. User Experience-oriented BDaaS Architecture(UE-BDaaSA)

Data Service Applications (DSA): DSA is the UE-BDaaSA entry for all users. DSA can handle data retrieval (keyword search and semantic retrieval), data analysis, and data visualization, and other service requests.

In the UE-BDaaSA, the module designed to support the user experience is User Entity & User Behavior Library (UEUBL) of DSE. UEUBL would be responsible for not only user management but motoring and modeling user behavior. The inclusion of user entities leads to making connections between data source, applications, and users, while user behavior recorded provides optimize suggestions when evaluating user requests, all this process made the architecture user-oriented.

4 Classification and BDaaS request processing

BDaaS can be divided by functions into such types: processing, retrieval, analysis, mining, visualization, services, etc., and each type of service is independent. When processing the user requests multiple services (the same type or different types) will be composition in light

of to the actual needs, the process of some typical requests is described below.

4.1 Processing request for BDaaS

Under certain specific situations, user has to receive data in different context, such as on the different terminates (PC and mobile phones), different bandwidth limitations and different operating systems. While returning the data for the request, the format of the returned data should be changed to be adaptive with the user's context and make sure that the data can be explained exactly and efficiently.

Therefore, the request of BDaaS will be processed as follows:

STEP1. Receives the user's retrieval request Q_P and the user's context description C_P;

STEP2. For the specific data request, get the target data *Rap* and a set of transformation rules provided by the environment $\Psi=\{\varphi_1,\varphi_2...\varphi_n\}$;

STEP3. Based on the user's context description C_P, we choose φ_i from Ψ which means that by transforming *Rap* with φ_i rule the returned data can be transformed exactly and efficiently;

STEP4. Transforms the target data $Rap \xrightarrow{\varphi_i} Rap^*$;

STEP5. Return *Rap**.

4.2Analysis request for BDaaS

For big data analysis services, service interfaces will decomposed analyze request into many sub-requests, and distributed to multiple analytical BDaaS, then each sub-analysis perform analysis request to corresponding data source respectively, finally, analyze results were assembled in the data service interface and returned to the user.

Analysis scenario is unlike retrieval scenario, although the analysis results maybe not clear, but the target data source and analysis rules are clear respectively. Therefore, the data request should contain the target data sources and analysis rules.

Analysis request of BDaaS will be processed as follows:

STEP1. User generate data analysis request Q_A which contain the target data sources $d_1,d_2,\ldots d_n$, format requires Fs, analysis rule set $Rule_1=\{r_1, r_2,\ldots, r_m\}$ and analysis results items *I*;

STEP2. Based on target data source in request Q_A, services matching component locate the required data analysis services $\{ S_1, S_2, \ldots, S_k \}$ from the data service registry center;

STEP3. According to user requests Q_A, services composition component generate services composition rules $Rule_2$ and data results assembly rules $Rule_3$;

STEP4. With $Rule_2$ and $Rule_3$, request processing component decompose Q_A to sub requests set $Q_{Asub}=<Q_{A1}, Q_{A2},...,Q_{Ak}>$ which is corresponding target services set $\{ S_1,S_2,\ldots S_k \}$;

STEP5. Rule decomposition components decompose Rule1 into several disjoint subsets, such as $\{r_1, r_3\},\{r_2,r_4,\ldots,r_m\}$;

STEP6. Sub-analysis request QAsub and sub rule sets will sent to target data sources $\{ S_1, S_2,\ldots, S_k \}$ respectively;

STEP7. $S_1, S_2,\ldots, S_k$ execute the sub requests in parallel, and temporary results of the analysis R_{A1}^*, $R_{A2}^*, \ldots, R_{Ak}^*$ were obtained;

STEP8. With $Rule_3$, *Fs, I*, get the results R_{items} by assembling $R_{A1}^*, R_{A2}^*, \ldots, R_{Ak}^*$;

STEP9. Output R_{items}.

4.3 Visualization request for BdaaS

Many mature open source visual component libraries provide a great convenience for big data visualization service. The goal of big data visualization services is that, for the user demand of data visualization, provides users with visual scripting leveraging existing visualization component libraries. Big data visualization service currently supports both popular D3.js and Fusioncharts chart library. As some users may be more familiar with certain types of script, the user can specify the desired output script type in the visualization request.

Before visualization, data should be given, and these data may be the result of user's retrieval request or analysis request. So the processing of big data visualization request can be summarized as, perform big data retrieval or big data analysis service first, then input the result data to the visual data service, finally output visual scripting or web scripts containing visual scripts to user.

Therefore, visualization request of BDaaS will be processed as follows:

STEP1. Receives the user's retrieval request G_D, visualization graphics requirements G_G and output script type T_S;

STEP2. Execute Q_s and Q_A extracted from G_D in terms of the processes of big data retrieval service or analysis service, then obtain the temporary result set Rs^* or R_A^*;

STEP3. Generating the format rule set $Rule_1$ from G_G;

STEP4. Matching the temporary result set Rs^* or R_A^*, and generating the visualization script G_O with T_S;

STEP5. Services matching component select the matching visualization big data service S_1 in the data service registry center;

STEP6. Input Rs^* or R_A to S_1, perform visualization tasks, and output visual scripts G_O and S_1.

5 Discussion and Comparison

Existing data service architecture were designed primarily for data CRUD operation, however UE-BDaaSA was designed to provide data processing, analysis and visualization services based on shielding the complexity of the data resources and operations.

This paper contrasted UE-BDaaSA and WCF DS, OSDI three services architecture in data object, data model data types, data sources, semantic, service description, service, operation, etc., as shown in Table 1. It can be seen from the table, each of architectures has their advantage, and UE-BDaaSA mainly for the big data service, which supports unstructured data model and semantic technologies, and provides thorough described services model and big data-oriented application. Especially UE-BDaaSA provides support for unstructured data, and provides a wide variety of service, such as analysis and visualization services. Therefore, UE-BDaaSA is a practical architecture and exactly what the big data age needs.

Table I The comparison among UE-BDaaSA, WCF DS, and OSDI

Architecture / Attributes	WCF DS	OSDI	UE-BDaaSA
user preferences	Not consider	Not consider	Based on user preferences
data model	OData	SDO.NET	E-R, K/V, Column-Oriented, Document Oriented, unstructured data model(such as GDM)
data type	Structured data, part of unstructured data	Structured data, part of unstructured data	Structured data, semi-structured data, unstructured data
support semantic or not	no	no	yes
service description	no	no	Contains the service content, the content of the data source and the user entity, etc
Service mode	static	static	Build service dynamically according to customer's request ,on-demand services
service operation	CRUD of Data	CRUD of Data	obtain the original data or results by querying, analysis, visualization services
request form	OData-based Http URI	XQuery	requests such as Keywords retrieval, semantic retrieval (SPARQL), visualization and analysis
format of operating results	JSON/ATOM	unknown	At least JSON, XML, D3 script, Fusioncharts Script

6 Conclusions

So far the research on BDaaS is not yet mature. This article put forwards a clear definition of BDaaS, designed a experience-oriented BDaaS architecture UE-BDaaSA, and described the requests processing flow of retrieval model, analytic and visualization in detail. UE-BDaaSA is a logical architecture which provides users with processing, analysis, and visualization services, by shielding the difference heterogeneous data sources and its operation, as well as composed kinds of BDaaS dynamically according to user behavior and application requests. Comparing with traditional data services framework, UE-BDaaS has many advantages, such as the ability to support unstructured data model, supports semantic, with more detailed service model, supports for multiple operations and service mode, it is consequently what the big data age needs and a innovative mode to help people to gain intrinsic value from big data more easier.

Acknowledgments

This work is supported by the National Key project of Scientific and Technical Supporting Programs of China (Grant No. 2012BAH01F02,2013BAH10F01,2013BAH07F02); the National Natural Science Foundation of China(Grant No.61072060).

References

[1] Lohr S. The age of big data[J]. New York Times, 2012, 11.
[2] Global Big Data-as-a-service Market 2012-2016[EB/OL].
[3] http://www.technavio.com/content/global-big-data-service-market-2012-2016
[4] Greenplum[EB/OL].http://www.greenplum.com
[5] Isilon[EB/OL].http://www.isilon.com/
[6] AWS Marketplace[EB/OL].https://aws.amazon.com/marketplace
[7] Windows Azure MarketPlace[EB/OL].http://datamarket.azure.com/
[8] Google BigQuery[EB/OL]. https://cloud.google.com/products/big-query
[9] SnapLogic[EB/OL]. http://www.snaplogic.com/
[10] Carey M J, Onose N, Petropoulos M. Data services[J]. Communications of the ACM, 2012, 55(6): 86-97.
[11] Mackey A. Windows Communication Foundation[M]//Introducing. NET 4.0. Apress, 2010: 159-173.
[12] Oracle and/or its affiliates. Oracle Data Services Integrator 10gR3(10.3), [EB/OL].http://docs.oracle.com/cd/E13162_01/odsi/docs10gr3/datasrvc/Data%20in%20the%2021st%20Century.html, 2011