

Data Organization In Restful Web Services Response Using Versioning (Build Number) & Structuring

Mitul Mandanka¹, Nehal Rajput²

^{1,2}Dept of Computer Engineering

^{1,2}Swarnim Startup And Innovation University

Abstract- *Rest-Application Programming Interfaces (REST - APIs) play an important role in mobile apps development as well as web application development, especially in mobile devices that have less processing power and capacity of storage. In the current scenario, mobile and web apps have to interact with the database to retrieve, store and perform various operations and transactions through the REST-APIs. By using REST-APIs mobile applications will easily connect with central heterogeneous system networks. Representational State Transfer (REST) has gained widespread acceptance across the web as a simpler alternative to WSDL - Web Service Description Language and SOAP - Simple Object Access Protocol interfaces are easier-to-use, resource-oriented models to expose their services. In this paper, we present a concept of response structure of any APIs response through JSON and versioning(application build number) to manage various problems faced by software mobile application developers like crash application, data fetch problem, data miss-match problem, etc.*

Keywords- REST-APIs, SOAP, REST, Web Services, JSON, Versioning (Build Number) & Structuring

I. INTRODUCTION

The best approach to manage APIs for circled heterogeneous structures contrasts from the commendable contraptions as offered by Apache Maven, which is expected to develop structure measures for the get-together of creatively and homogeneously relative ancient pieces. when we are using REST APIs then we have no constraints related to the platform dependency and by doing that user has no restriction to sending a request over the heterogeneous server and gets a response to that particular request. [1]

Restful APIs are the bridge between request send and heterogeneous servers. They communicate via a network environment such as. Internet Or Intranet. Restful Web Services are platform-independent and provide flexibility to send requests from software applications which are running on different platforms.

In a recent scenario, All software and mobile applications are modified/updated their functionality day by day and When we have any new requirement for updation in the current system then the developer needs to do changes on Database, APIs and Mobile applications and need to release new versions. There is a problem when the user can't update the mobile app from Google play store and Apple AppStore and by doing that user and developers face lots of issues such as data miss-match, application crash and displaying wrong data to apps, etc.

To resolve these problems, We can do modification on APIs response structure by including build numbers response on particular version. If we follow that structure then we can reduce server storage occupied by APIs as well as create standard response structure so mobile developers can easily integrate that in their Android, iOS Mobile Applications and other public platforms.

Application Programming Interfaces (APIs) have assisted organizations with coordinating applications or even form new heterogeneous systems. There is a pattern to cultivate the development and adaptation of new ways for request-response through APIs, In new industry requirements have several usages for REST API. Many web clients are now using REST API to fetch & update web data to the server and stored data in the database. [2][3]

II. WEB ORIENTED REQUEST AND RESPONSE ARCHITECTURE FOR WEB SERVICES

Web platforms are the kinds of web programming that utilizes normalized informing conventions over the dispersed network. In recent scenarios, Web Services are widely used in business for a website, mobile apps and other web devices. it's also developing technology for researchers. Web Services mainly designed for the transfer of web data between client and server. Web services used many protocols to achieve their functionality like SOAP, HTTP, XML, JSON, etc. SOAP (Simple Object Access Protocol) to send XML/JSON data between applications and that data is sent over HTTP. XML and JSON are dataformat which contain data and provide metadata around it. There are several

platforms and programming languages where we can develop web services like Java, .NET, Python, Node.js, etc. [4]

When the Web Services are deployed on the server with the registered URL. Web clients are able to send requests and get a response. We can use XML OR JSON for sending requests and getting responses via HTTP methods such as GET, POST, PUT, DELETE, etc. Web Service provides good network services with multiple Web Business, Mobile Application, Web Platforms, Web Clients. Using Web Services we can help all types of businesses like big organizations, small and medium-sized companies and individuals to take their business to the next level by providing software web services to those businesses. In recent scenarios using the Internet, we have a huge set of various communication web devices to access resources via send request to the web server. [4]

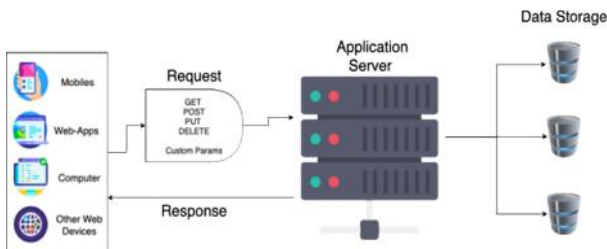


Fig.1 Web Services Request-Response Architecture

The Key Terminology of Web Services are defined below,

- Request and Response: In the web services, We have to pass our input as a request and output is generated as a response.
- Message Exchange Format: Format that can be used for sending a request and getting output in response. There are 2 formats that are used for message exchange formats, which are XML and JSON.
- Service Provider or Server: Web services that are hosted on the Server.
- Service Consumer or Client: A service consumer is an actual web user that can use these web services.
- Service Definition: Service Definition: Service definition is the protocol between service provider or server and service consumer. Service definition is the format and structure of request and response and endpoint.

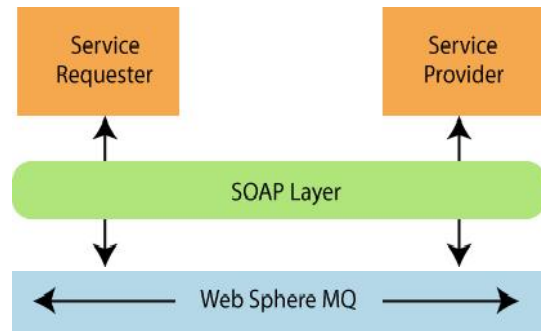


Fig.2 Service called through Transport [4]

Transport: Transport: How service call and All details of that service call is defined by transport. MQ - SeMessage Queue and HTTP are used for service calls. We can call the services on the internet by typing the URL of that particular service. When a request arrives at the service provider, it will take a request, do processing on that request and generate a response, and return the generated response into Message Queue. The requesters will get a response from the message queue so data exchange between requesters and service provided through the queue.

III. SPECIFICATION OF WEB SERVICES

3.1 SOAP

SOAP specification includes that the WSDL and UDDI are part of SOAP. It is utilized now without WSDL and UDDI. SOAP messages are hard-coded or produced without the utilization of a repository. SOAP commonly uses HTTP, but other protocols such as Simple Mail Transfer Protocol (SMTP) may be used. SOAP can be used to exchange complete documents or to call a remote procedure. [5]

3.2 REST

REST (Representational state transfer) is an architectural style consisting of a coordinated set of architectural constraints applied to web-components, web-connectors, and web-data elements, within a distributed system. In a recent scenario, REST is very easy and simpler in a style that makes it more accepted than SOAP. It is likewise less verbose with the goal that less volume is sent when imparting.. We can call REST-compliant systems RESTful systems. REST has principles such as, Separate the concerns of client and server, Stateless, Cacheable, Uniform interface, Layered system, and Code on demand. [5]

3.3 UDDI

When the Web Services are published on a web server (register in UDDI) and its interface is able to identified, described, discovered and then invoked by other WebApplications. Web Services connect one web device to another web device with each other via WWW or Server running on your local environment. Webservices give good connectivity for web based components, devices and web applications. Web servers are listening to requests at specific port over the web network and providing responses in form of json,html, xml,text, Images. [4]

3.4 WSDL

The WSDL structures the reason for the first Web Services detail. The accompanying figure shows the utilization of WSDL. At the left is a specialist co-op. The privilege is to help users. [5]

IV. CAPABILITIES AND OPPORTUNITIES OF TECHNOLOGIES FOR API VERSIONING

The operating API system has brought various difficulties to many product manufacturers, engineers, and engineers. Apart from the fact that many methods were suggested to find compelling ways to manage change: proven to be broken, it all helped to find a common weight associated with organizations as the number of integration APIs increased. A brief introduction to all new features associated with state-of-the-art broadcasting, such as Hypertext Transfer Protocol (HTTP), Uniform Resource Locator (URL), Domain Name System (DNS), and Uniform Resource Identifier (URI) need to understand their critical volume, as each of them forms the basis of a different strategy. [1]

4.1 SEMANTIC VERSIONING

The term semantic forming (semver) refers to a diagram of objects of different weights to educate designers about the level in relation to the variations that affect the various interpretations on the sides called from applications. One route to have three different numbers to adapt to major, minor, and related fixes, as developed by [7], the editor of GitHub.

While critical digital editions show invalid changes, both minor and adjustments are referred to as variable variables, with the additional distinction of additional minor bug fixes and the use of the correction rate. [7] Thus, semantic formation helps to assess the change to be made when the development of a different alternative while violating the changes is only allowed in the larger object.

4.2 VERSIONING WITH DNS

The use of subdomains, one of all types, allows help to have multiple versions in harmony. The overall structure, blocks the road-building components, is given by https://<version>. <domin_name>. <domain_ extension > A model would be:

projectapi-v1.progragon.com

and

projectapi-v2.progragon.com

Apart from that, there are significant inefficiencies in the use of subdomains to implement installed APIs, not least on the grounds that DNS crashes after compromise, but there is also a need to change firewall rules to ensure network between customers and employees as well. Another disadvantage is the elasticity effect, as it often influences at least one different end point that consistently depends on the same delivery cycle, as a rule that shows no purpose. [1]

4.3 VERSIONING WITH URL AND URI

In contrast to the DNS-based approach, where the identifier is usually more likely to capture it with a route-based pattern, the URI method provides more flexible methods for managing translation indicators at various stages in the backend, Java, category or method for example.

Depending on its position along the roads, the level determines the hypothesis of the rate at which it falls under the same output. Undoubtedly, the URI-based approach proves to be very profitable compared to DNS. The following example gives the middle granularity, so everything behind this API becomes part of the same release cycle:

https://<host>/projectapi/{versionIdentifier}/endpoints

When it comes to decision-making, the benefits of fine-grained API versions with a small portion of the unmodified or unstructured components included and its administrative costs need to be weighed against the effects of deceptively cut outings with a high number of unmodified components and related administrative costs.

V. DATA ORGANIZATION IN API RESPONSE WITH VERSIONING (BUILD NUMBER)

We define structure for api's responses where we have below data when we make requests with versionNumber

1.1.0 and buildNumber 1.0.1. In dictResponse and arrResponse data will be changed as per your api request.

```
{
  "status": true,
  "strStatus": "OK/Error",
  "strMessage": "Data retrieved successfully",
  "dictResponse": {
    "id":108,
    "firstName": "ABC",
    "lastName": "XYZ",
    "postsCount":8,
    "userName":"abc_xyz",
    "description":"Hi, I am programmer",
  },
  "arrResponse" : [],
  "versionNumber" : "1.1.0",
  "buildNumber" : "1.0.1"
}
```

In software products we have continuous updates and once we have minor updates then using this structure no need to change version, by changing buildNumber we can fulfill this requirement. We will change only the build number on request and in response we get data related to that particular buildNumber.

```
{
  "status": true,
  "strStatus": "OK/Error",
  "strMessage": "Data retrieved successfully",
  "dictResponse": {
    "id":108,
    "firstName": "ABC",
    "lastName": "XYZ",
    "postsCount":8,
    "userName":"abc_xyz",
    "Bio" : "Hi, I am programmer",
    "Followers":539,
    "Following":1120,
    "Website": "",
    "Category":"Entrepreneur",
    "Email":"abc_xyz@zyx.com",
    "Phone":"+919999999999"
  },
  "arrResponse" : [],
  "versionNumber" : "1.1.0",
  "buildNumber" : "1.0.1"
}
```

We define structure for api's responses where we have below data when we make requests with versionNumber

1.1.0 and buildNumber 1.0.1. In dictResponse and arrResponse data will be changed as per your api request.

VI. CONCLUSION & FUTURE WORK

Application Programming Interfaces (APIs) assist organizations by applying or creating new facilities. There is a pattern of developing and improving API management, which is not understood in many places. The frameworks developed through communication connections not only to take full advantage of businesses, determining the piece of Industry 4.0, in particular, attractive production lines but also in addition to the world of money adhering to Open Banking orders.

Gathering the web management needed by customers as their disclosure is an important issue in the IoT environment. Later, the development of home robots, light urban communities, smart cars and e-wellbeing in those IoT application spaces could be built. There are a large number of gadgets that can find movement that takes place including providing a variety of services to end customers. Most IoT gadgets are equipped to detect environmental restrictions and yet do not have the knowledge to provide a formal response based on the data obtained. From now on, it is important that IoT gadgets should be grouped together. Clients should be reminded of the existence of the group in order to benefit from the services offered by it, and from now on there is a need for a tool to expose the group. As the use of force is an important limitation of group disclosure. After that, a lightweight disclosure part of the group is required. [8]

REFERENCES

- [1] REST-API Versioning Strategies, [Online]. Available: <https://dzone.com/articles/rest-api-versioning-strategies>
- [2] Marr, B., Data Strategy: How to profit from a world of big data, analytics, and the internet of things, 2017, Kogan Page, Kindle Books
- [3] Passi, L. F., 'An open banking ecosystem to survive the revised Payment Services Directive: Connecting international banks and FinTechs with the CBI Globe platform', Journal of Payments Strategy & Systems, 12(4), pp. 335–345. 2018.
- [4] A Comprehensive Study on Web Services Basics, International Conference on Recent Developments in Science, Humanities & Management-2018 (ICD SHM-18), ISSN: 2319-8354
- [5] JyotiL.Khachane,LatikaR.Desai."A survey paper on web services in IOT".IJSR.Vol 4.issue 12,December 2015.
- [6] Yan, Z., Li, H., Zeadally, S., Zeng, Y. & Geng, G., Is DNS Ready for Ubiquitous Internet of Things?, IEEE

- Access, Access, IEEE, p. 28835. 2018. doi: 10.1109/ACCESS.2019.2901801
- [7] Preston-Werner, T., Semantic Versioning 2.0.0, [Online]. Available: <https://semver.org>.
- [8] Survey Paper on Web Services in IOT, Jyoti L. Khachane¹, Latika R. Desai, International Journal of Science and Research (IJSR), ISSN (Online): 2319-7064, Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611
- [9] A Comprehensive Study on Web Services Basics, International Conference on Recent Developments in Science, Humanities & Management-2018 (ICD SHM-18), ISSN: 2319-8354.
- [10] Triveni Mishra, Gaurav Raj, "QoS implementation in web services selection and ranking using data analysis". 2017 7th international conference on cloud computing, data science and engineering confluence. 537-542. 2017 IEEE.
- [11] W3C, Hypertext Transfer Protocol -- HTTP/1.1, 1999. [Online]. Available: <https://www.w3.org/Protocols/rfc2616/rfc2616.html>
- [12] Zhang, Lili, et al. "Research on IOT RESTful Web Service Asynchronous Composition Based on BPEL." Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2014 Sixth International Conference on. Vol. 1. IEEE, 2014.
- [13] Gao, Ruiling, et al. "Web-based motion detection system for health care." Computer and Information Science (ICIS), 2015 IEEE/ACIS 14th International Conference on. IEEE, 2015.
- [14] Rambold, Michael, et al. "Towards autonomic service discovery a survey and comparison." Services Computing, 2009. SCC'09. IEEE International Conference on. IEEE, 2009.
- [15] Rong, Wenge, and Kecheng Liu. "A survey of context aware web service discovery: from user's perspective." Service Oriented System Engineering (SOSE), 2010 Fifth IEEE International Symposium on. IEEE, 2010.
- [16] Wu, Hang, and Chaozhen Guo. "The research and implementation of Web Service classification and discovery based on semantic." Computer Supported Cooperative Work in Design (CSCWD), 2011 15th International Conference on. IEEE, 2011.
- [17] Guinard, Dominique, et al. "Interacting with the soa-based internet of things: Discovery, query, selection, and on-demand provisioning of web services." Services Computing, IEEE Transactions on 3.3 (2010): 223-235.
- [18] Im, Janggwan, Seonghoon Kim, and Daeyoung Kim. "IoT mashup as a service: Cloud-based mashup service for the internet of things." Services Computing (SCC), 2013 IEEE International Conference on. IEEE, 2013.
- [19] L. Atzori, A. Iera, and G. Morabito, The Internet of Things: A Survey, Elsevier Computer Networks, 2010.