SharePoint REST API - Improve application performance

Just by modifying the REST call header, you can achieve performance improvement in your application. For the SharePoint developers, REST is a very widely used API. We will go through the analysis to understand what modifications we can make and how we can improve page render.

**JSON Light support in REST SharePoint API**

[JSON Light](https://www.microsoft.com/en-us/microsoft-365/blog/2014/08/13/json-light-support-rest-sharepoint-api-released/) is an [open standard](https://en.wikipedia.org/wiki/Open_standard) that provides developers the option so that they can decide on how much metadata is returned just by modifying the REST call header. You can use this REST API header in

* SharePoint 2013 SP 1
* SharePoint Online
* SharePoint 2016 and onwards

**Where is the performance issue in REST Call?**

In the header of the REST call we pass, "Accept": "application/json;odata=verbose". **Accept header** actually shows what exactly you need in your response when you get data back from the server.

Please go through the screenshot below. It shows what data we get from the server. I haven’t even covered all the data in the screenshot that we got from the REST call.

You can see the response comes with so much data that you may not need and which **increases the extra load of the call**.



**How does it improve the REST Call performance?**

In the Accept header, odata comes with an **additional two options**. You can set these two options according to your requirements. We will take a look into all the scenarios and check what response size we get from each option.

I have made a GET REST call on one custom SharePoint list and created **three different observations**. For each of the requests, I have mentioned the **Fiddler trace**. Please go through the section below.

**"Accept": "application/json;odata=verbose"**

We have been using this option. Will get the maximum amount of data creating the heaviest payload.

**Total bytes received in response -** 7925

odata verbose fiddler trace

**"Accept": "application/json;odata=minimalmetadata" OR "Accept": "application/json”**

By-default the value of odata is minimalmetadata so if you don’t specify anything, the call will consider **odata=minimalmetadata**. We will get a moderate amount of data in return of the request.

**Total bytes received in response** - 5900

odata minimalmetadata fiddler trace

**"Accept": "application/json;odata=nometadata"**

We will get the least amount of data by this call. You can check your response and if it fits your requirement, you can go for this option to optimize the performance to the fullest.

**Total bytes received in response** - 5319

odata nometadata fiddler trace

**Conclusion:**

We went through all the options of **Accept Header** in the REST API call in order to achieve performance for the SharePoint application.