

SAP NW Gateway Performance Best Practices

David Freidlin



Agenda

- **Dos**

- Do use *JSON format*
- Do use GZIP compression
- Do use OData query options (such as *delta query, paging, \$count etc.*)
- Do use *\$expand and \$batch*

- **Don'ts**

- **SAP NW Gateway Server explicit implementations**

- **Backend Consumption recommendations**

Dos - JSON format & GZIP

➤ Do use *JSON format* to reduce consumer and SAP NetWeaver Gateway server time and in addition reduce network traffic.

❖ Example: `http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/BookingCollection?$format=json`

➤ Do use *GZIP* to compress data.

✓ Call SAP NW Gateway services with request header: *Accept-Encoding: gzip, deflate*

Dos - OData Query Options

Delta Query

- Do use *delta query* in case the end users trigger the same query calls with the same options very often.

This will fetch only the data from the SAP Business Suite that was created/changed/deleted since the end-user last asked.

- ❖ Example: By querying the information for the first time, SAP NW Gateway will generate a delta token at the end of returned data:

Query: `http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/Bookings?$filter=Bookdate%20eq%20datetime'2008-12-30T00:00:00'`

End of SAP NW Gateway response:

```
</entry>
<link rel="delta" href="Bookings/?$filter=Bookdate%20eq%20datetime'2008-12-30T00:00:00'?!deltatoken='22F9DE51B07CD00DE10000000A1A51DD_20130714070353'"/>
</feed>
```

By accessing the same query next time with fetched delta token, SAP NW Gateway will return only the created and changed data:

Query: `http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/Bookings?$filter=Bookdate%20eq%20datetime'2008-12-30T00:00:00' &!deltatoken='22F9DE51B07CD00DE10000000A1A51DD_20130714070353'`

Dos - OData Query Options

\$select and \$filter

➤ Do use *\$select* to specify the fields returned in the result set.

The value of \$select specifies that a response from an OData service should return a subset of the properties of the collection of entries.

❖ Example: You have to fetch a list of Flights and Flight Details by retrieving the data from the same FlightCollection for both use cases. For list you would like to present only 2 values (carrid & connid). However, for Flight details you would like to fetch all properties of specific flight.

For Flights list (header data) you fetch only 2 properties carrid & connid of each entry as following:

`http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/FlightCollection?$select=carrid,connid`

For Flight details you fetch all the properties of the specific Flight as following:

`http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/FlightCollection(carrid='AA',connid='0017',fldate=datetime'2010-10-20T00%3A00%3A00')`

➤ Do use *\$filter* to fetch a subset of the entities in the entity set.

The \$filter system query option identifies a subset of the entries in the collection.

Where \$select is a column filter of the result set, \$filter is a row filter.

❖ Example: The following query will fetch only entries where filter=carrid eq 'LH' and connid eq '0400':

`http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/FlightCollection?$filter=carrid eq 'LH' and connid eq '0400'`

Both filters (\$select & \$filter) can be combined in order to reduce the result set.

`http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/FlightCollection?$filter=carrid eq 'LH' and connid eq '0400' &$select=carrid,connid`

Dos - OData Query Options

Client & Server paging

➤ Do use client-side paging *\$top & \$skip* to reduce the returned items in the result set.

❖ Example: The following query will fetch only first 10 Flights.

`http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/FlightCollection?$top=10`

➤ Do use *server-driven paging \$skiptoken* to fetch a subset of the entities in the entity set.

The server can limit the amount of entries sent by default as response in order to avoid a performance overhead if a large number of entries is requested. A next link in the payload contains the skiptoken set by the server in the response context.

❖ Example: The following Query will return 20 entries (out of 91) and in addition the skiptoken value.

Query: `http://services.odata.org/Northwind/Northwind.svc/Customers`

End of response:

```
</entry>
<link rel="next" href="http://services.odata.org/Northwind/Northwind.svc/Customers?$skiptoken='ERNSH'" />
</feed>
```

Once the client requests the same query next time with fetched skiptoken, the server will return the next subset of the entries (20 customers starting after **'ERNSH'**).

Query: `http://services.odata.org/Northwind/Northwind.svc/Customers?$skiptoken='ERNSH'`

Dos - OData Query Options

\$count & \$inlinecount

➤ Do use *\$count* when only the count of entries is needed without the queried entries.

❖ Example: The following query will fetch only the count of FlightCollection entries:

```
http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/FlightCollection/$count
```

➤ Do use *\$inlinecount* to fetch the overall count of a collection of entries together with the queried entries.

Note: The count is the overall quantity of the entries in the collection after all \$filter system query options have been applied and not the count of the fetched entries.

❖ Example: The following query will fetch all the entries of the CarrierCollection and in addition will fetch the count:

```
http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/CarrierCollection?$inlinecount=allpages
</author>
<link href="CarrierCollection" rel="self" title="CarrierCollection"/>
<m:count>18</m:count>
```

Dos - OData Query Options

\$expand/deep insert

➤ Do use *\$expand/deep insert* for associated entity types when you need to fetch or create deep data.

- Consider to use *parallel calls* instead of *\$expand* in case of one association.
- Use *\$expand/deep insert* for more data which contains more than one association call.
- Implement the **basic** *\$expand* on SAP NW Gateway server (instead of the generic *\$expand* implementation).

\$expand/deep insert vs. single calls:

- *\$expand/deep insert* reduces roundtrips from the client to the SAP NW Gateway server as well as the SAP NW Gateway server to the SAP Business Suite. As a result, network time is reduced.
- *\$expand/deep insert* reduces SAP NW Gateway server response time and the consumption of resources.
- Basic *\$expand* implementation can reduce the SAP Business Suite times, because the number of database operations and the application logic can be reduced.

❖ Example: The following query will fetch all the entries of the FlightCollection and in addition associated entries of FlightCarriers:

`http://<host>:<port>/sap/opu/odata/IWFND/RMTSAMPLEFLIGHT/FlightCollection?$expand=FlightCarrier`

Dos - \$Batch

➤ Do use *\$batch* to send the data of several requests in a single HTTP request from the client to the SAP NetWeaver Gateway server for entity types that are not associated.

- Use parallel calls in case of two Read calls or two Update calls with *multiple* atomic units of work (different change sets). As an alternative to two parallel calls, you can consider using *\$batch*.
- Use *\$batch* in case of more than two calls without associated entity types (do not make too many client – SAP NetWeaver Gateway server roundtrips).

\$batch performance optimizations and considerations (compared to single calls):

- *\$batch* reduces the number of roundtrips between the end-user and the SAP NetWeaver Gateway server. As a result, network time is reduced.
- *\$batch* reduces the number of roundtrips from the SAP NW Gateway server to the SAP Business Suite to one in case of Update calls with a single atomic unit of work.
- *\$batch* reduces SAP NW Gateway server response time and the consumption of resources.
- *\$batch* reduces CPU time of the SAP Business Suite in case of Update calls with a single atomic unit of work.
- Consider the balance between the number of roundtrips and the payload size. A payload size which is too big can affect the network time due to bandwidth constraints.
- Consider using *\$batch* because it can have a negative impact on the response time since SAP NetWeaver Gateway calls SAP Business Suite/s in sequential order (in case of Read calls or Update calls with multiple atomic units of work).

Dos - \$batch and \$expand vs. parallel calls

➤ The following table can help to choose the required option between \$expand, \$batch and parallel calls.

Operation	# of calls from Client to Gateway	
	2 calls	2+ calls
Associated Read	\$Expand/ parallel calls	\$Expand
Read without association	parallel calls	\$Batch
Update with <u>multiple</u> atomic units of work	parallel calls/Batch	\$Batch
Update with <u>single</u> atomic unit of work.	\$Batch	

Don'ts

- Do *not* trigger more than two synchronous calls from the client to the SAP NetWeaver Gateway server per single user operation.
- Do *not* use \$batch for associated entity types. Use \$expand instead.
- Do *not* fetch more data than will be presented. Use OData query options to reduce the result set to fit data to be presented.
- Do *not* call more than two parallel calls per single operation. Use \$batch for more than two parallel calls. More than two parallel calls consume more server resources and network bandwidth.

SAP NW Gateway Server explicit implementations

The following functions should be implemented explicitly on SAP NW Gateway in order to be consumed by Gateway client and achieve better performance:

➤ Delta Query

More information: http://help.sap.com/saphelp_gateway20sp06/helpdata/en/3a/8e956126d741d183b93fa87f30c857/frameset.htm

➤ Server-driven paging (\$skiptoken)

More information: http://help.sap.com/saphelp_gateway20sp06/helpdata/en/99/29f0c9bfd04ea4bb60b084fd45d624/frameset.htm

➤ Basic \$expand.

SAP NW Gateway provides a default (generic) \$expand implementation, but it only represents a feature to provide functional completeness. Therefore, it is recommended to do a basic \$expand implementation in order to provide the possibility to the application to read and return entities deeply.

More information: http://help.sap.com/saphelp_gateway20sp06/helpdata/en/3a/8e956126d741d183b93fa87f30c857/frameset.htm

➤ \$filter and Client-Side paging (\$top and \$skip)

More information: <http://scn.sap.com/docs/DOC-43000>

➤ \$inlinecount

➤ **\$select.** There is a default implementation provided by SAP NW Gateway. However, it can be overridden explicitly by application in order to select only required values from SAP Business system.

Backend Consumption

- Use RFC/BAPIs on the Backend which support OData features (such as \$filter, paging, order by, etc.)
 - Implement filtering, sorting, paging “closer” to the Backend DB

- Use Backend BAPIs which are performance optimized for stateless usage.
 - BAPIs that are optimized for statefull usage might include pre-fetching of not yet used data in order to avoid the fetching of the data in next requests.
 - Use BAPIs without pre-fetching of not yet used data.
 - Do not use post processing which fills buffers in BAPIs that are mainly designed to be accessed in a later calls. In stateless environment (as Gateway) there are no later calls.

- Use BAPIs that fetch only the required information
 - Don't use BAPIs that fetch too much information.
 - Don't use BAPIs that fetch not enough information, which require additional BAPIs calls.

- Use mass enabled BAPIs for mass data retrieval from backend system/s.
 - Do not use non mass enabled BAPI in a loop.
 - Do not use BAPIs that consist out of a loop over non mass enabled BAPIs (fake mass enablement).

Additional Information

SAP NW Gateway documentation:

http://help.sap.com/saphelp_gateway20sp07/helpdata/en/3a/8e956126d741d183b93fa87f30c857/frameset.htm

Open Data Protocol (OData):

<http://www.odata.org/>

Blog:

<http://scn.sap.com/community/netweaver-gateway/blog/2013/07/09/gateway-performance-best-practices--how-to-consume-gateway-from-performance-point-of-view>



Thank you

© 2013 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

National product specifications may vary.

These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark> for additional trademark information and notices.

© 2013 SAP AG. Alle Rechte vorbehalten.

Weitergabe und Vervielfältigung dieser Publikation oder von Teilen daraus sind, zu welchem Zweck und in welcher Form auch immer, ohne die ausdrückliche schriftliche Genehmigung durch SAP AG nicht gestattet. In dieser Publikation enthaltene Informationen können ohne vorherige Ankündigung geändert werden.

Einige der von der SAP AG und ihren Distributoren vermarkteten Softwareprodukte enthalten proprietäre Softwarekomponenten anderer Softwareanbieter.

Produkte können länderspezifische Unterschiede aufweisen.

Die vorliegenden Unterlagen werden von der SAP AG und ihren Konzernunternehmen („SAP-Konzern“) bereitgestellt und dienen ausschließlich zu Informationszwecken. Der SAP-Konzern übernimmt keinerlei Haftung oder Gewährleistung für Fehler oder Unvollständigkeiten in dieser Publikation. Der SAP-Konzern steht lediglich für Produkte und Dienstleistungen nach der Maßgabe ein, die in der Vereinbarung über die jeweiligen Produkte und Dienstleistungen ausdrücklich geregelt ist. Keine der hierin enthaltenen Informationen ist als zusätzliche Garantie zu interpretieren.

SAP und andere in diesem Dokument erwähnte Produkte und Dienstleistungen von SAP sowie die dazugehörigen Logos sind Marken oder eingetragene Marken der SAP AG in Deutschland und verschiedenen anderen Ländern weltweit. Weitere Hinweise und Informationen zum Markenrecht finden Sie unter <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark>.