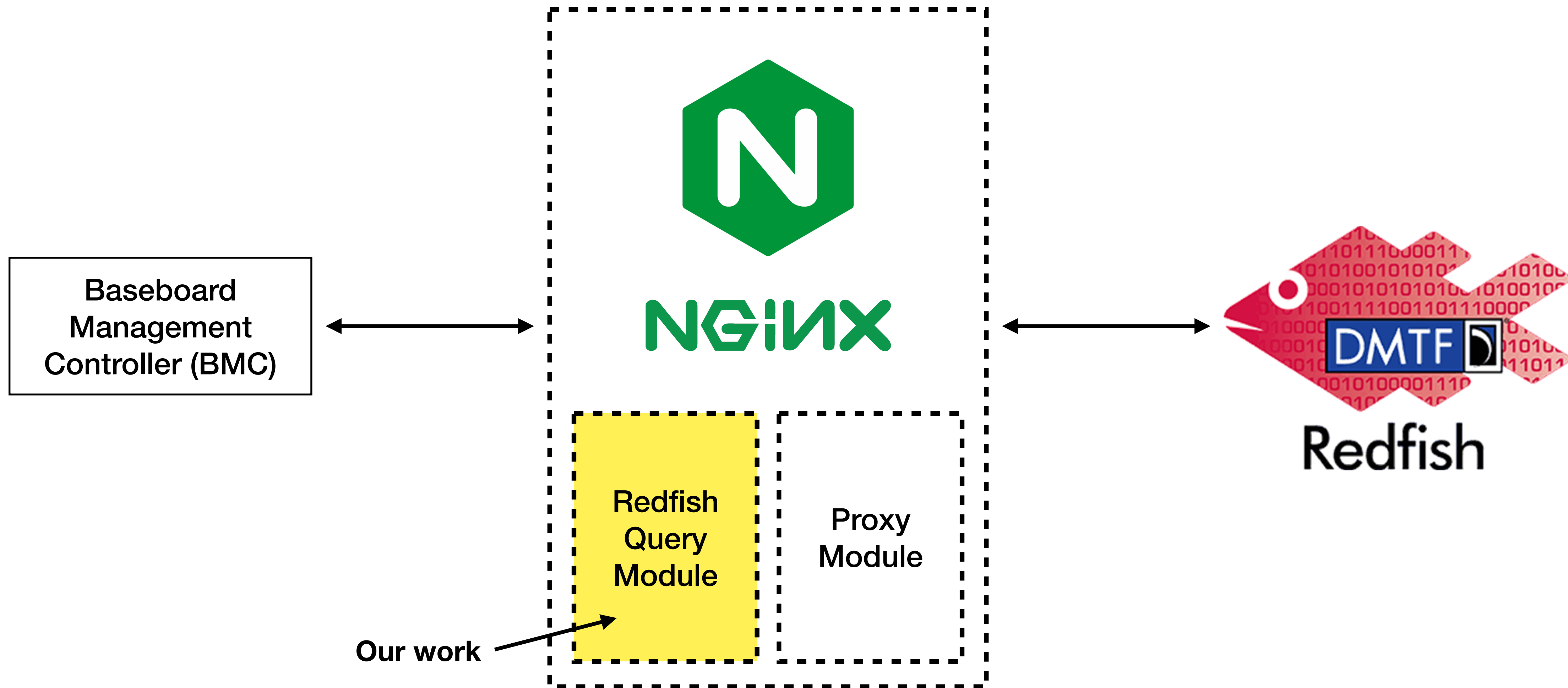
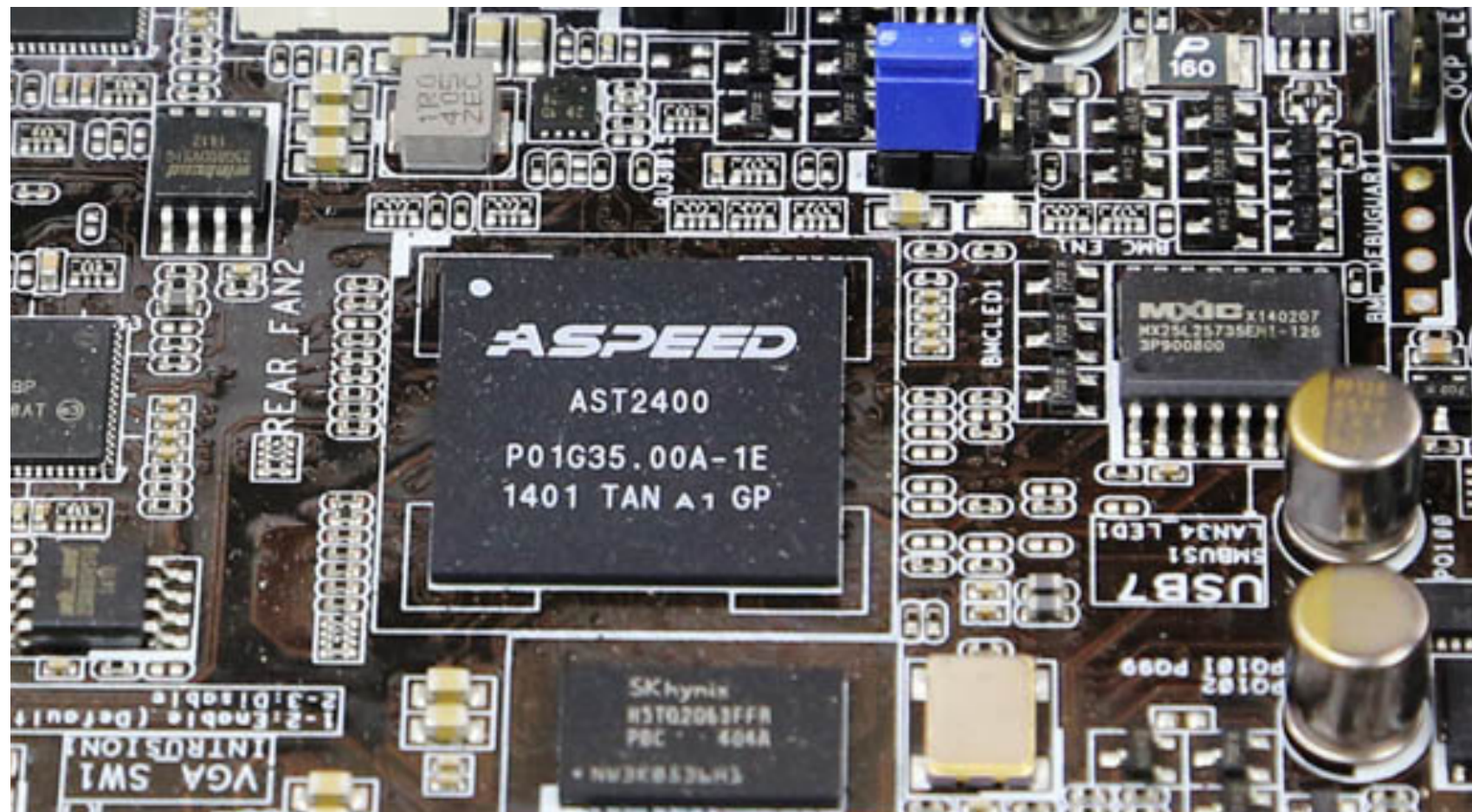


Prototype of Configurable Redfish Query Proxy Module

Chanyoung Park, Yoonsue Joe, Myounghwan Yoo, Dongeun Lee, Kyungtae Kang*
Hanyang University, XSLAB, Texas A&M University-Commerce

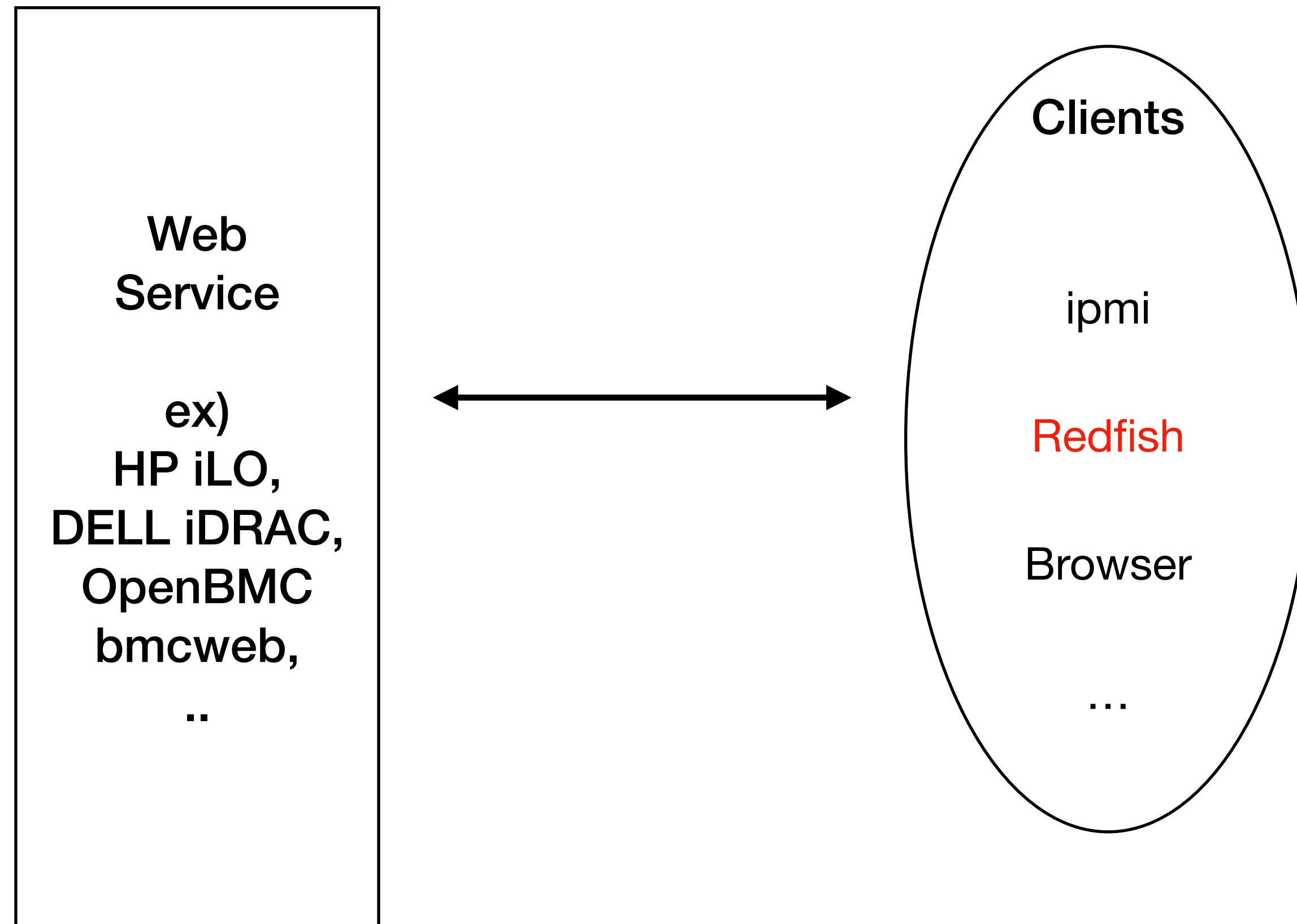


Background: What are BMC and Redfish?

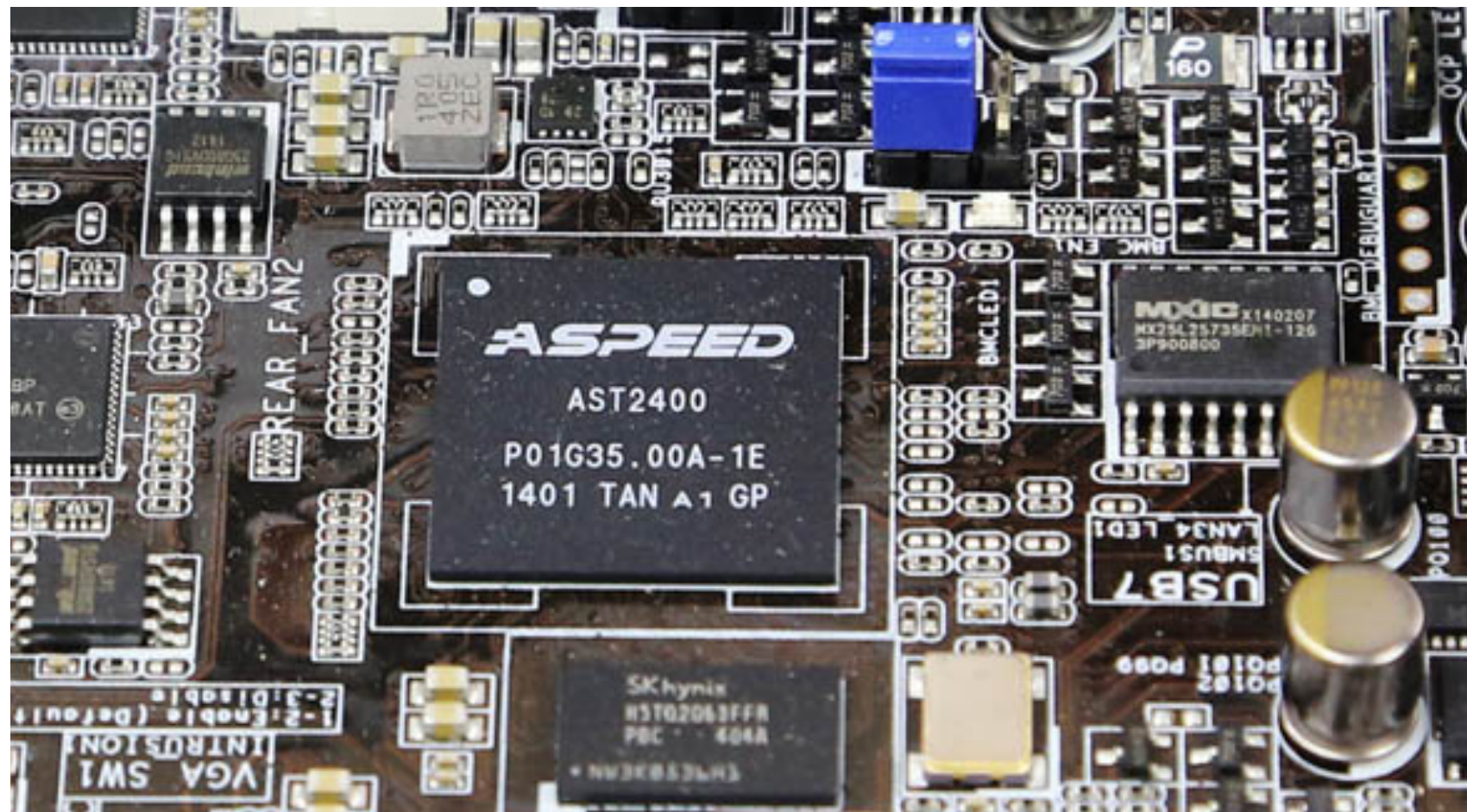


Provide server baseboard management things

- Power control
- Fan control
- OS status
- etc.

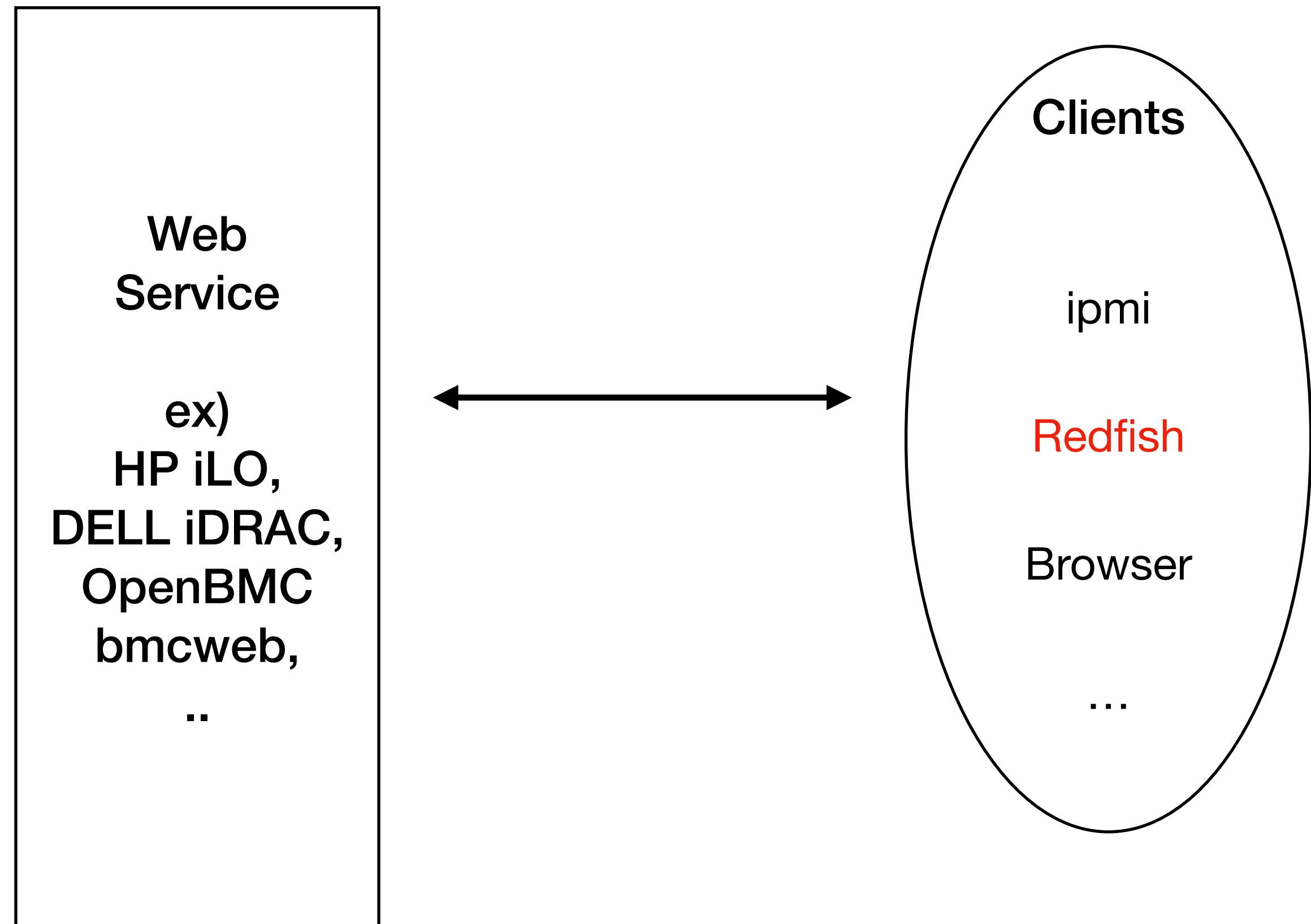


Background: What are BMC and Redfish?

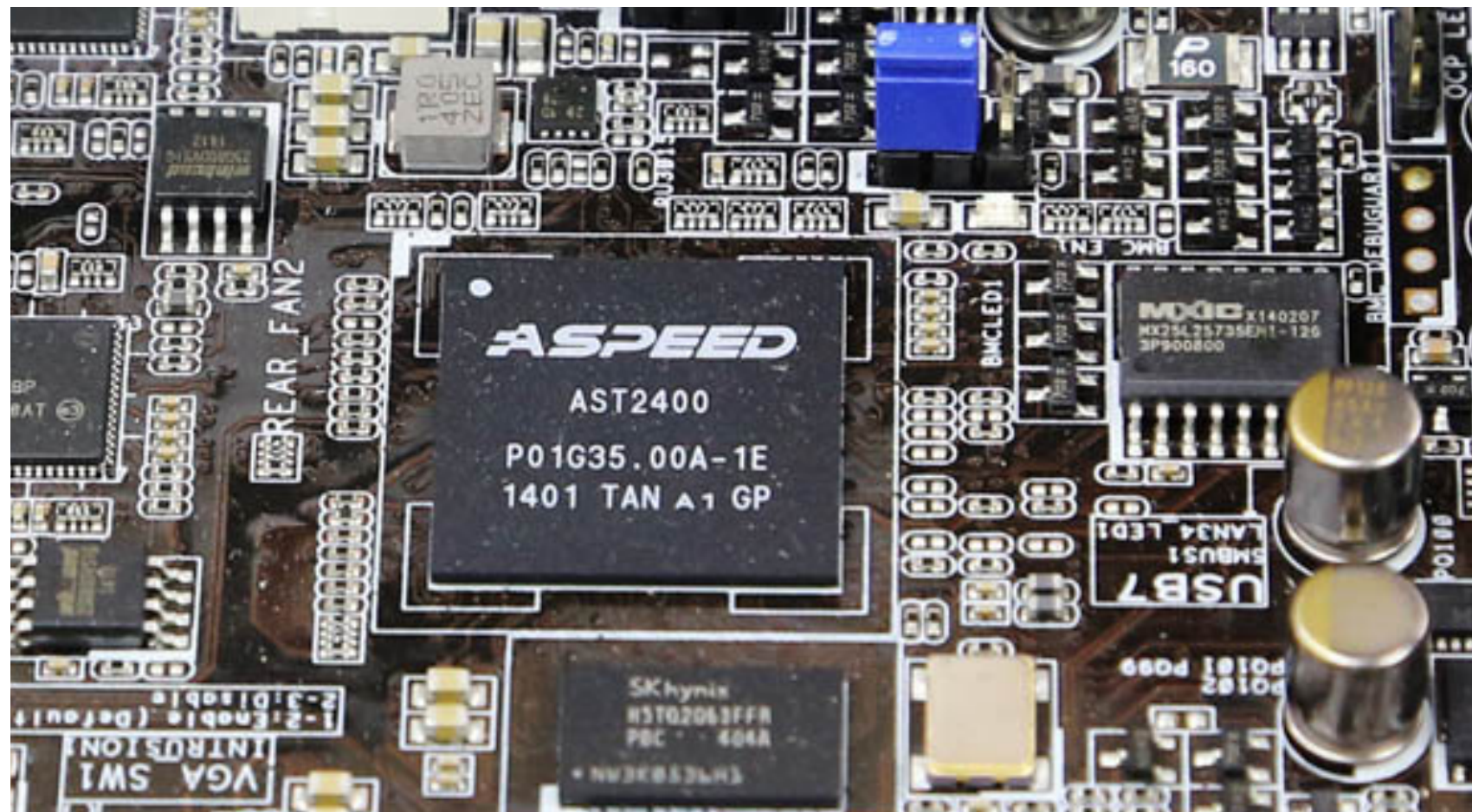


Provide server baseboard management things

- Power control
- Fan control
- OS status
- etc.

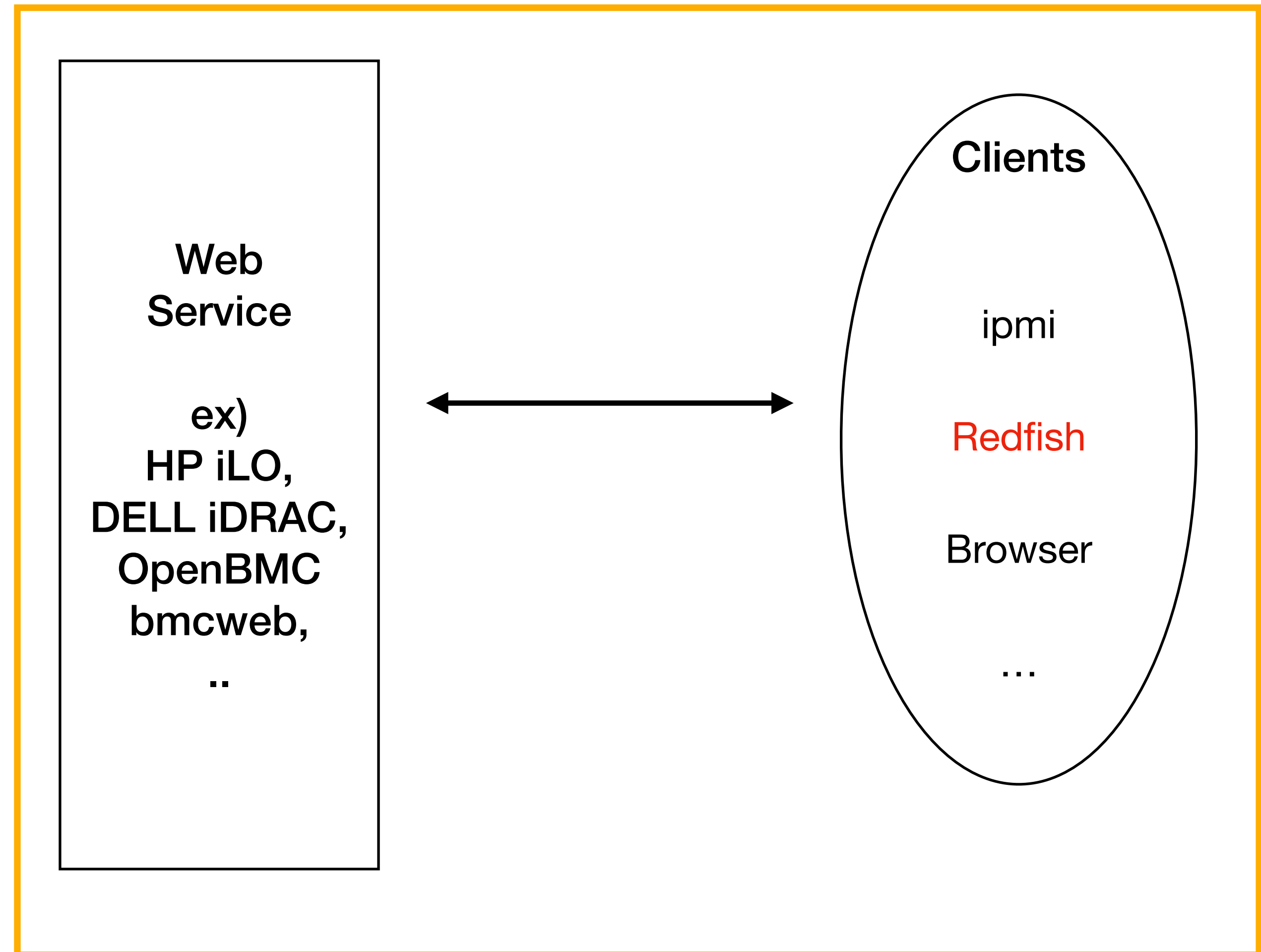


Background: What are BMC and Redfish?

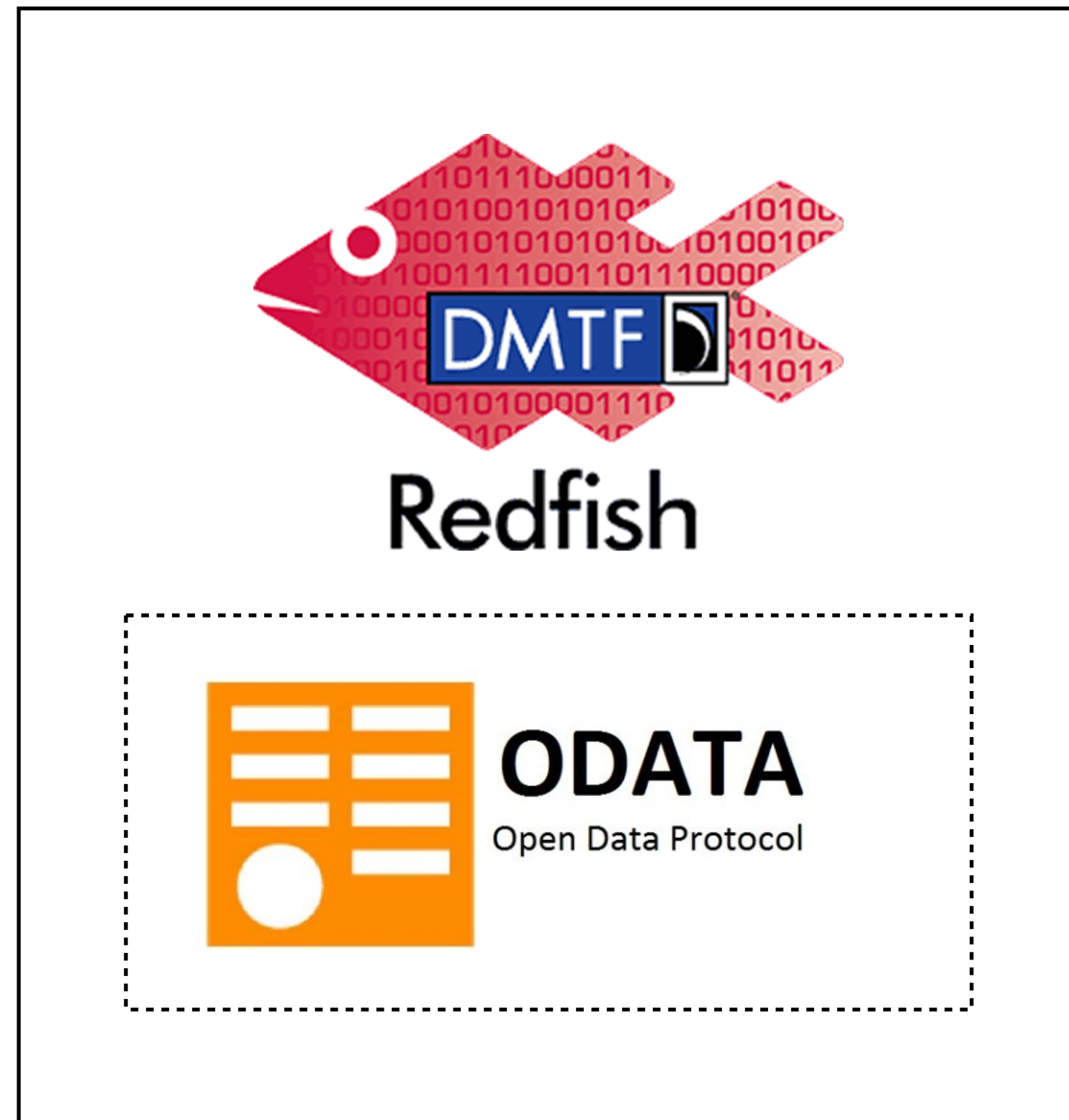


Provide server baseboard management things

- Power control
- Fan control
- OS status
- etc.



Background: What is Redfish Query?



Redfish is build on top of Open Data Protocol (OData).

- OData provides standardized client side query

 - ex) `http://url?$top=3`

 - `http://url?$expand&select=Name,SystemType`

- Redfish follows the format of OData,

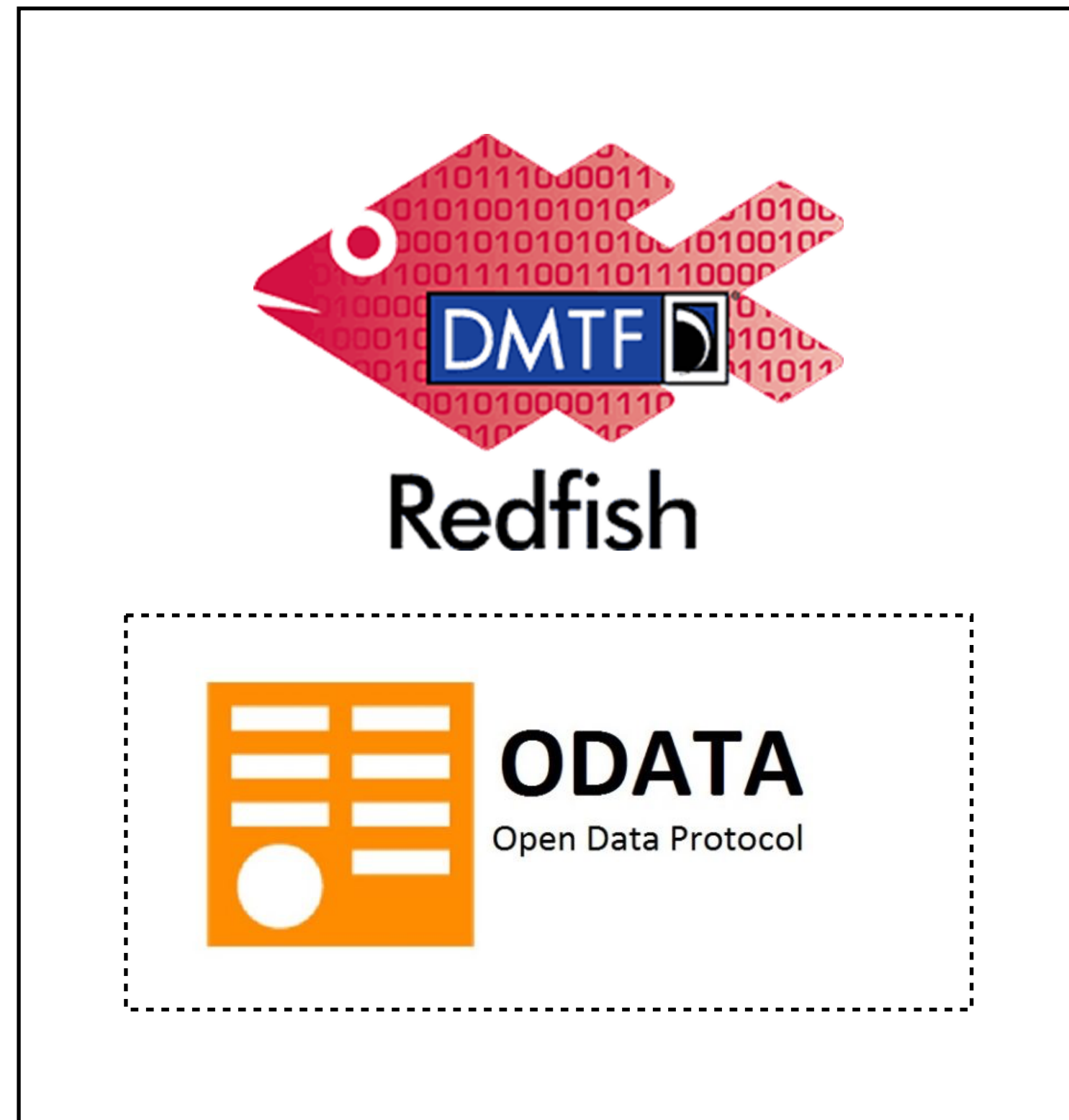
 - but to implement OData query features are **optional**

- Redfish suggests to support some query features

 - from OData and a couple of new query features

 - ex) `$top`, `$skip`, `$expand`, `$filter`, `$select`, `only`, `excerpt`

Background: What is Redfish Query?



Redfish is build on top of Open Data Protocol (OData).

- OData provides standardized client side query

 - ex) `http://url?$top=3`

 - `http://url?$expand&select=Name,SystemType`

- Redfish follows the format of OData,

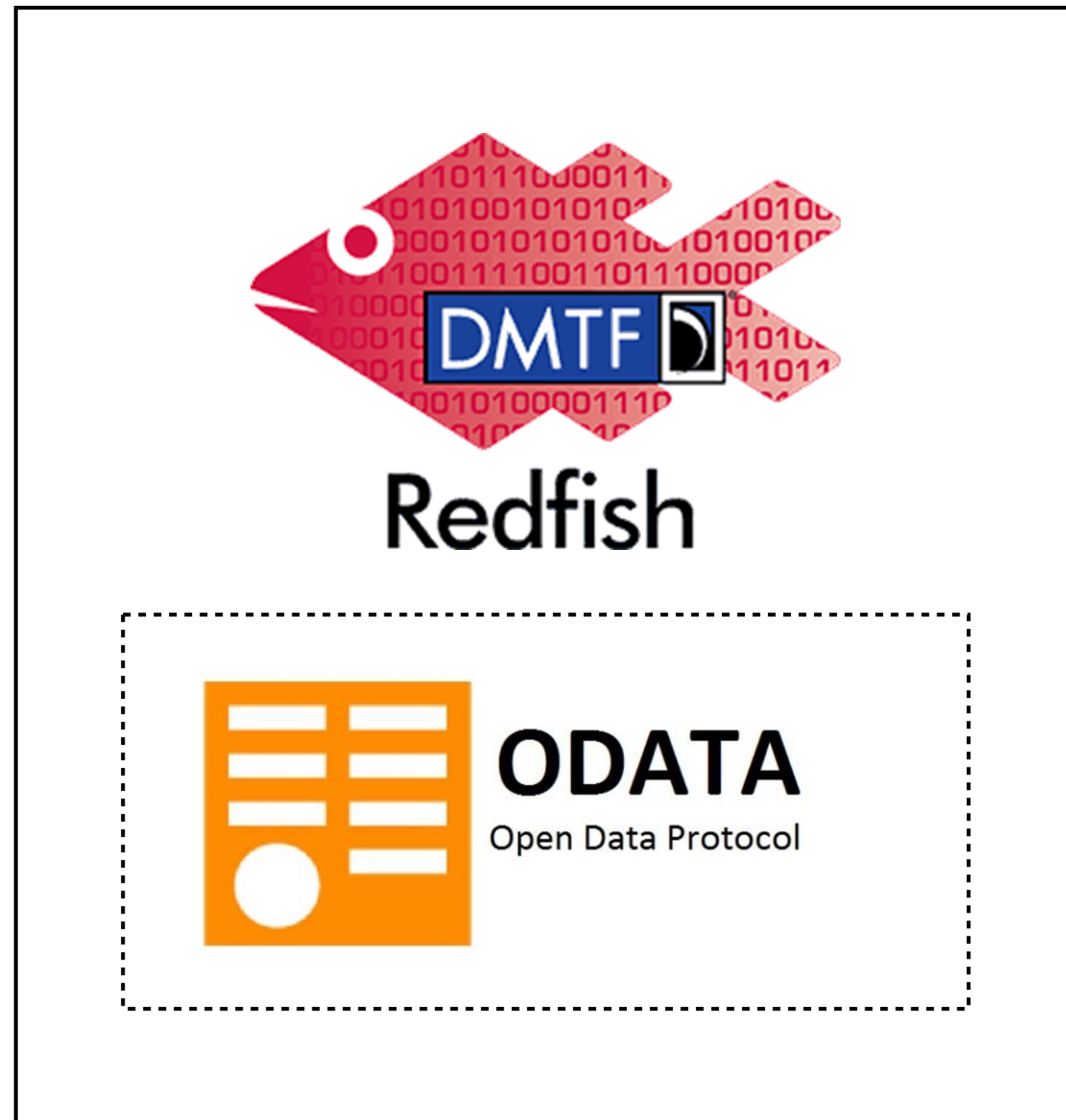
 - but to implement OData query features are **optional**

- Redfish suggests to support some query features

 - from OData and a couple of new query features

 - ex) `$top`, `$skip`, `$expand`, `$filter`, `$select`, `only`, `excerpt`

Background: What is Redfish Query?



Redfish is build on top of Open Data Protocol (OData).

- OData provides standardized client side query

 - ex) `http://url?$top=3`

 - `http://url?$expand&select=Name,SystemType`

- Redfish follows the format of OData,

 - but to implement OData query features are **optional**

- Redfish suggests to support some query features

 - from OData and a couple of new query features

 - ex) `$top`, `$skip`, `$expand`, `$filter`, `$select`, `only`, `excerpt`

What's Hard? And Our Approach

Vendor A: Let's support \$filter query in resource Chassis

Vendor C: Let's support \$select query in all resources

Vendor B: Let's support \$top, \$skip queries in resource Events

Vendor D: We have no plan to support any query features

Confusing...



Redfish

Why? Because implementing query features are not easy + time consuming.

- Developers must implement all query features in data layer by themselves, if they are not using RDB.
- OData ecosystem is mature in some platform and languages
- Redfish query features are **optional**

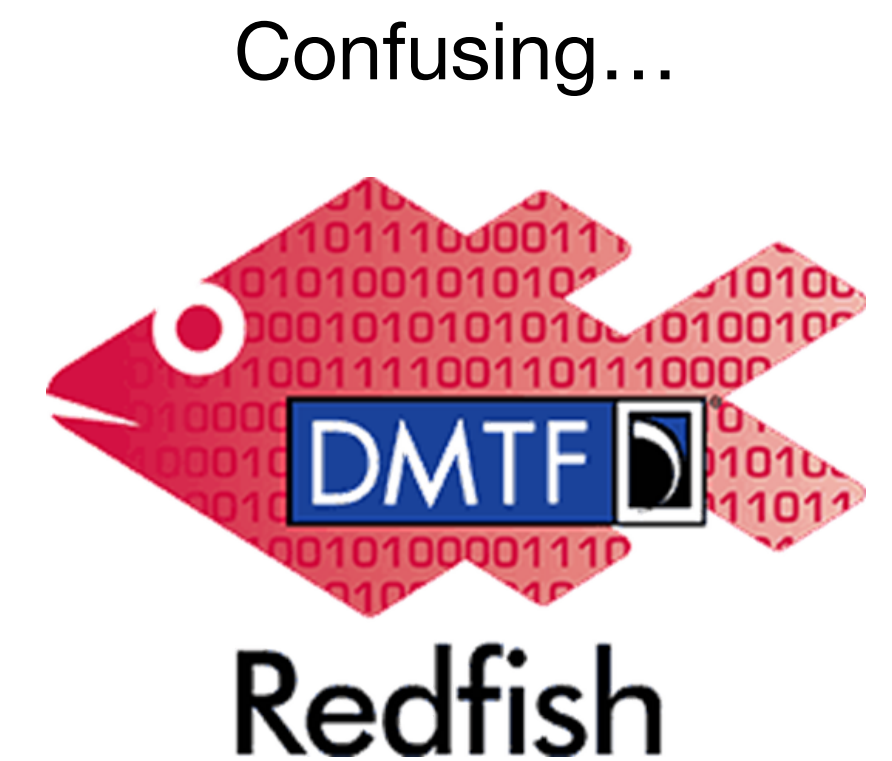
What's Hard? And Our Approach

Vendor A: Let's support \$filter query in resource Chassis

Vendor C: Let's support \$select query in all resources

Vendor B: Let's support \$top, \$skip queries in resource Events

Vendor D: We have no plan to support any query features



Why? Because implementing query features are not easy + time consuming.

- Developers must implement all query features in data layer by themselves, if they are not using RDB.
- OData ecosystem is mature in some platform and languages
- Redfish query features are **optional**

What's Hard? And Our Approach

Vendor A: Let's support \$filter query in resource Chassis

Vendor C: Let's support \$select query in all resources

Vendor B: Let's support \$top, \$skip queries in resource Events

Vendor D: We have no plan to support any query features

Why? Because implementing query features are not easy + time consuming.

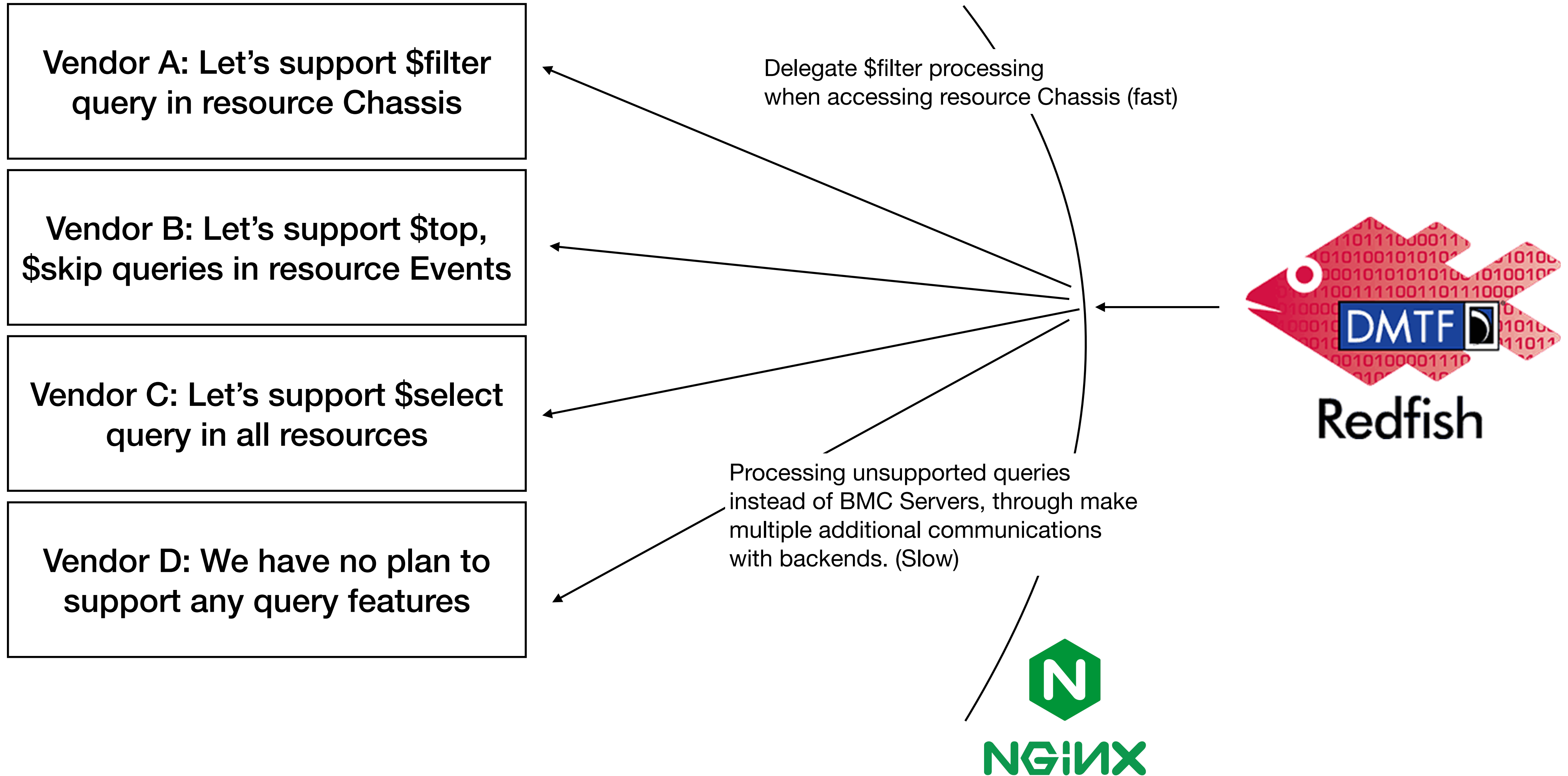
- Developers must implement all query features in data layer by themselves, if they are not using RDB.
- OData ecosystem is mature in some platform and languages
- Redfish query features are **optional**

Confusing...



Redfish

What's Hard? And Our Approach



What's Hard? And Our Approach

Vendor A: Let's support \$filter query in resource Chassis

Vendor B: Let's support \$top, \$skip queries in resource Events

Vendor C: Let's support \$select query in all resources

Vendor D: We have no plan to support any query features



Redfish

RedfishTool

Do similar things to some target resources, with some query features

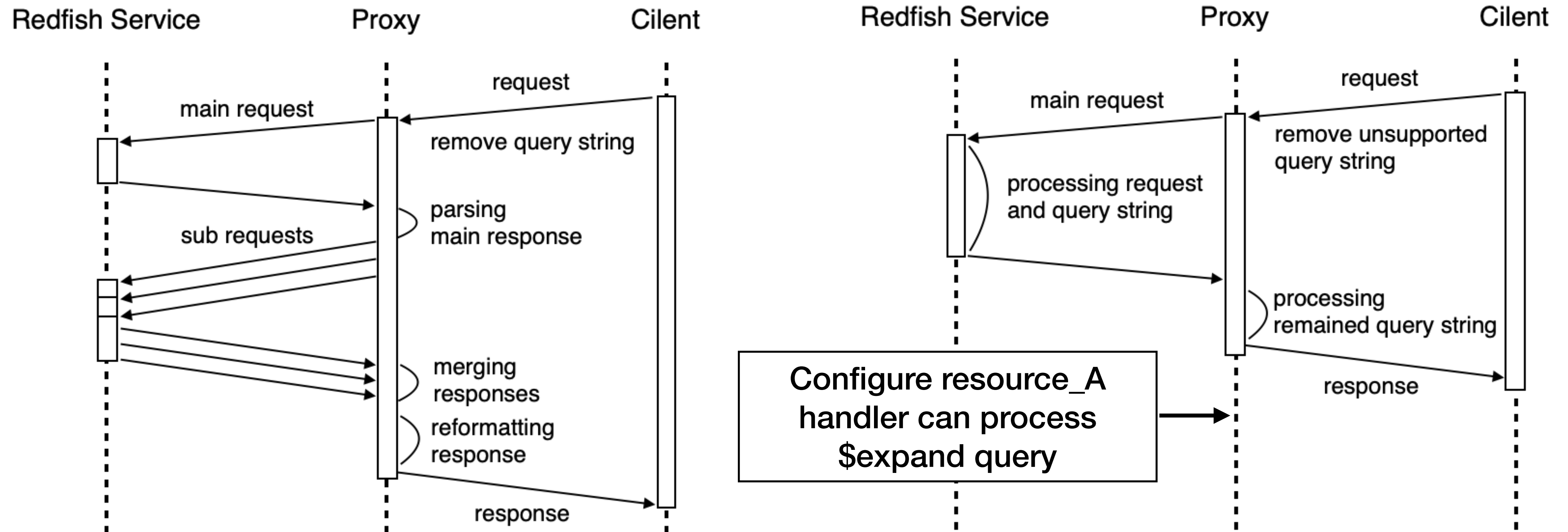
Difference with RedfishTool

1. We target all resources and query features.
2. We can leverage backend server's query processing by configuration.

Approach Details

Client Request:

`https://bmc-host/redfish/v1/resource_A?$expand=.$levels=1&$select=Name,Id&$top=10`



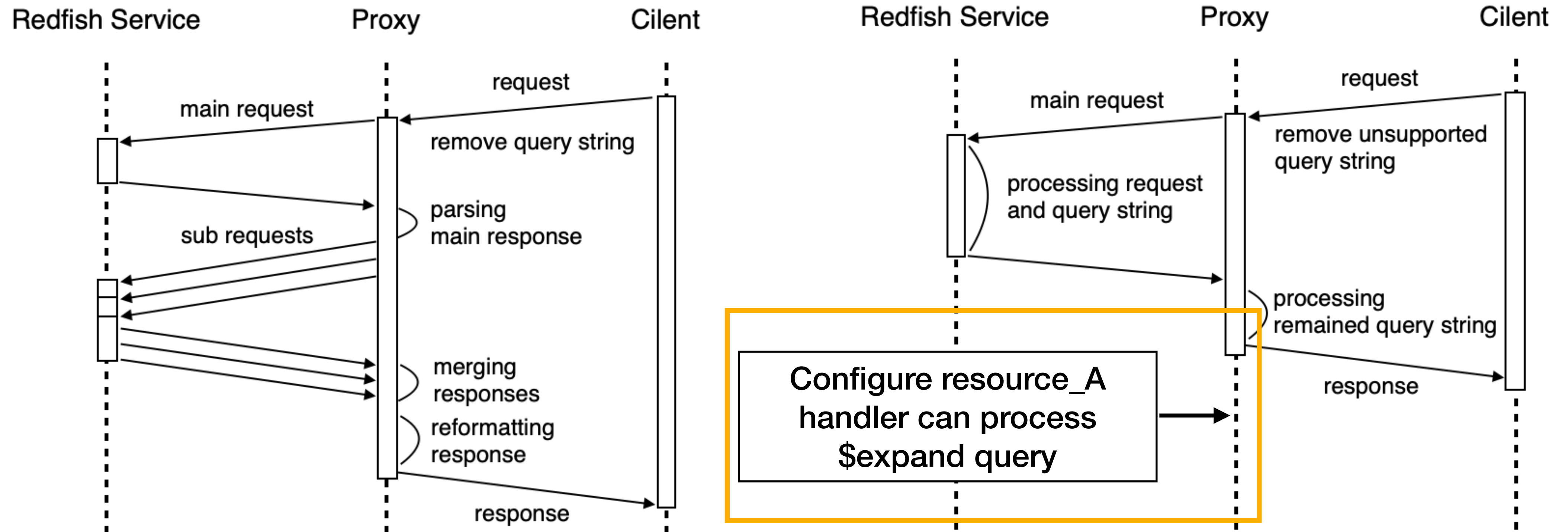
(a) Redfish service's target resource handler does not support any query.

(b) Redfish service's target resource handler supports \$expand query.

Approach Details

Client Request:

`https://bmc-host/redfish/v1/resource_A?$expand=.(levels=1)&$select=Name,Id&$top=10`



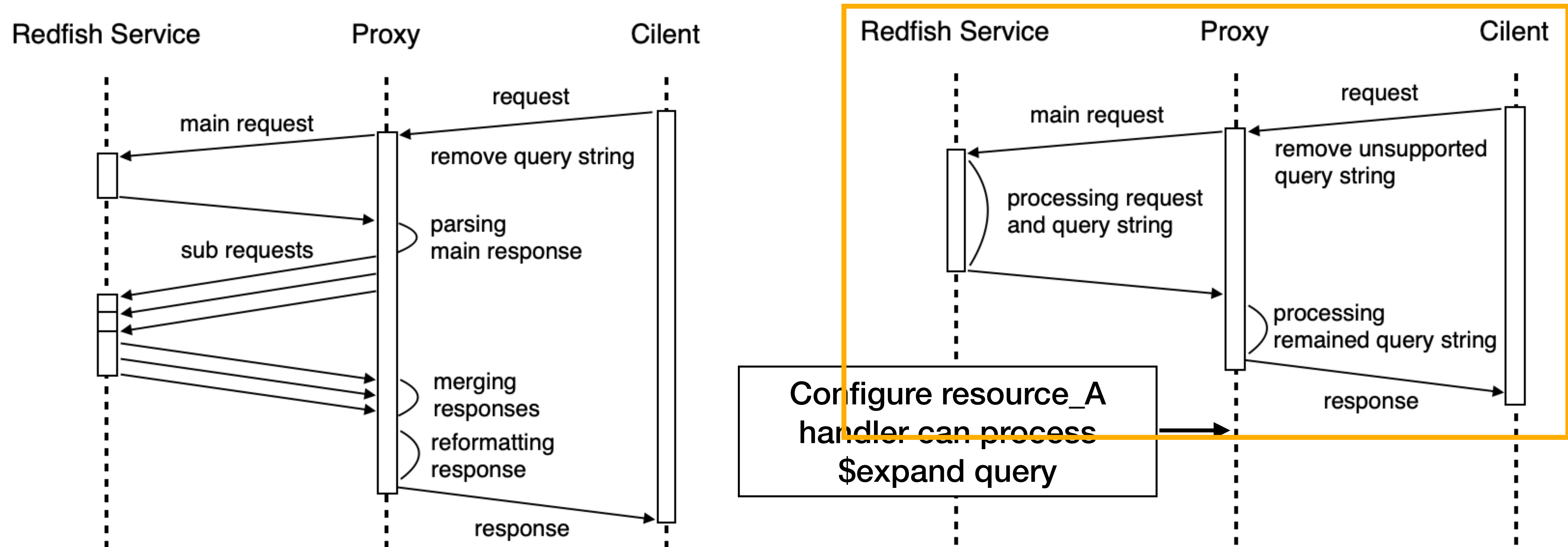
(a) Redfish service's target resource handler does not support any query.

(b) Redfish service's target resource handler supports \$expand query.

Approach Details

Client Request:

`https://bmc-host/redfish/v1/resource_A?$expand=.(levels=1)&$select=Name,Id&$top=10`



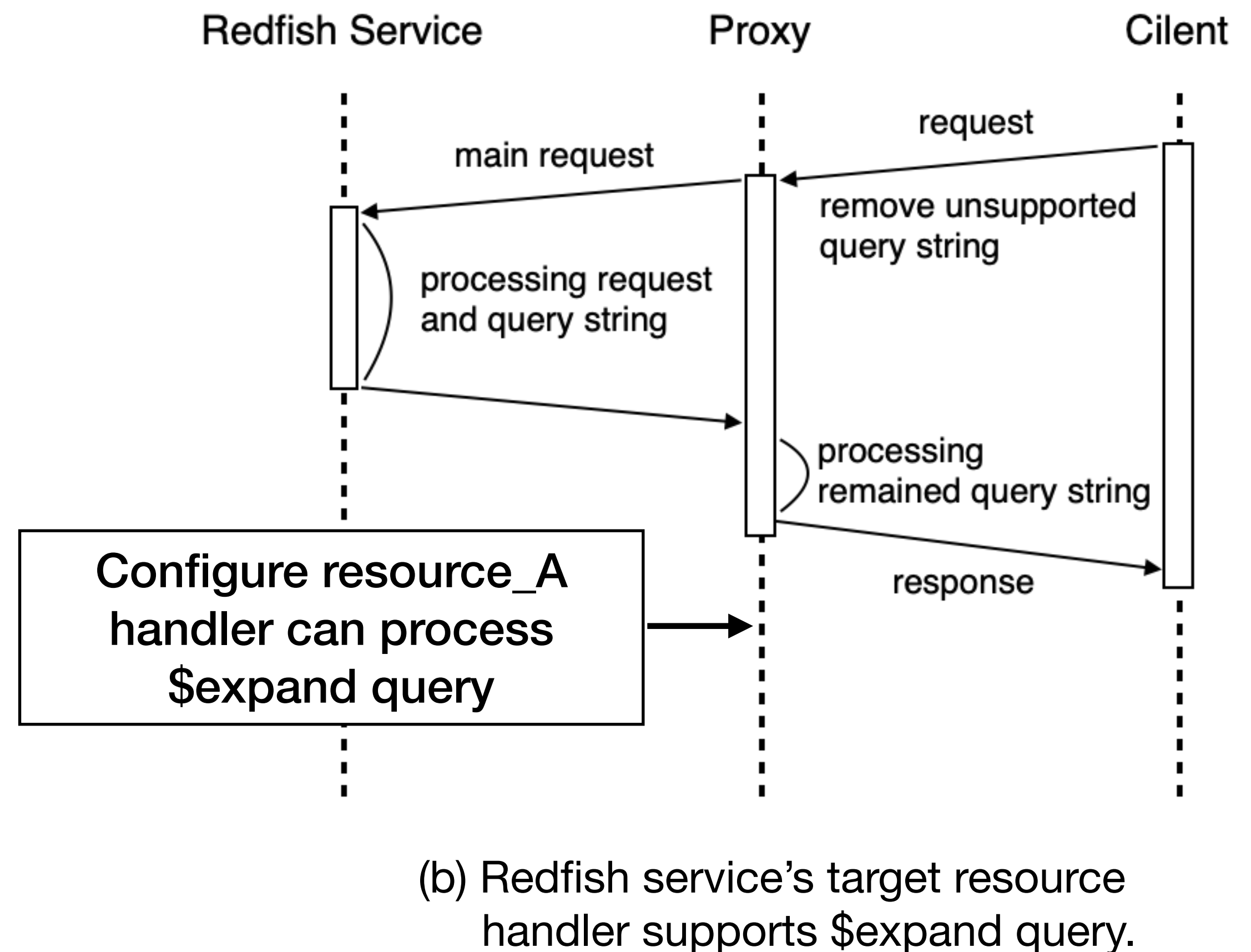
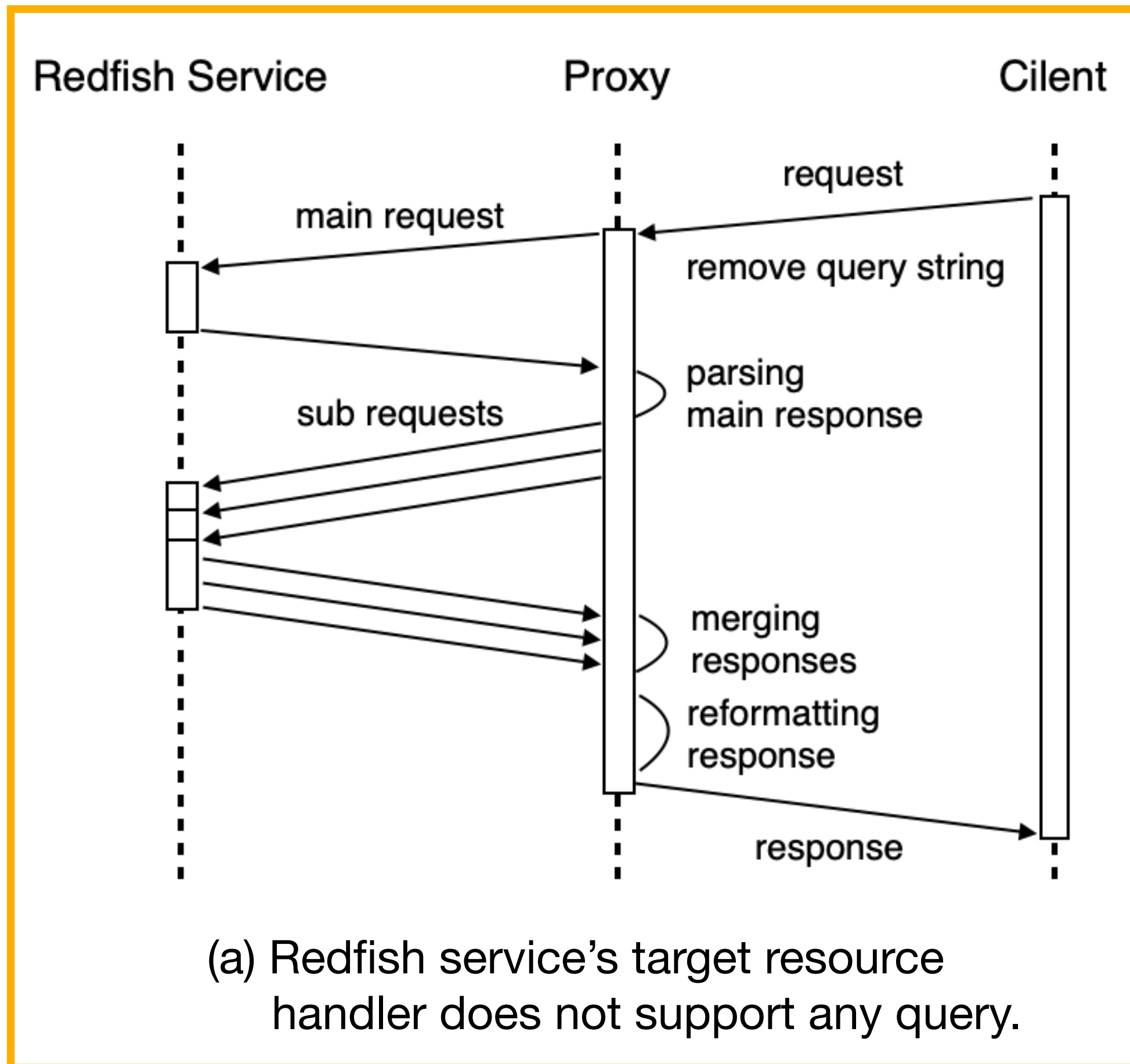
(a) Redfish service's target resource handler does not support any query.

(b) Redfish service's target resource handler supports \$expand query.

Approach Details

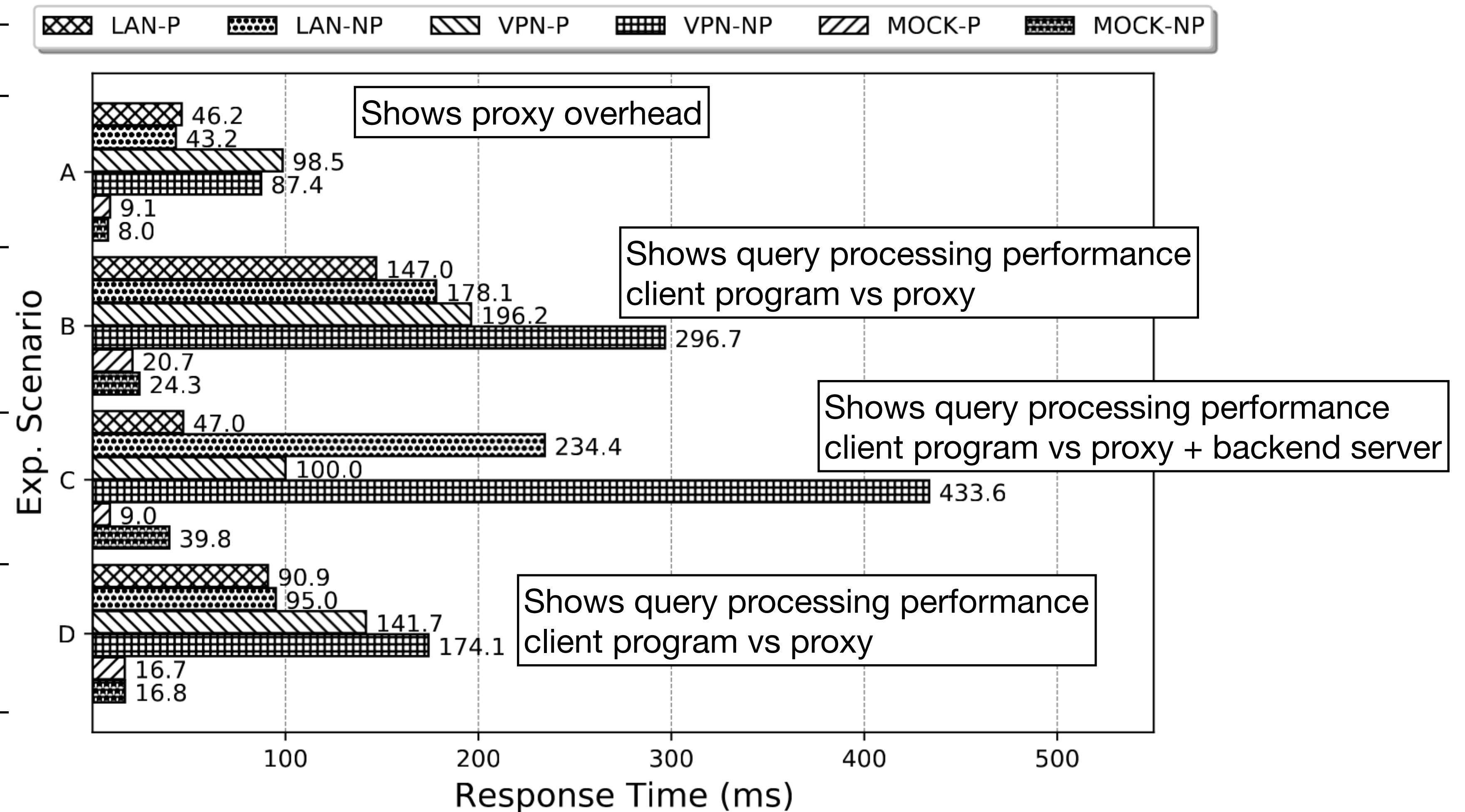
Client Request:

`https://bmc-host/redfish/v1/resource_A?$expand=.(levels=1)&$select=Name,Id&$top=10`



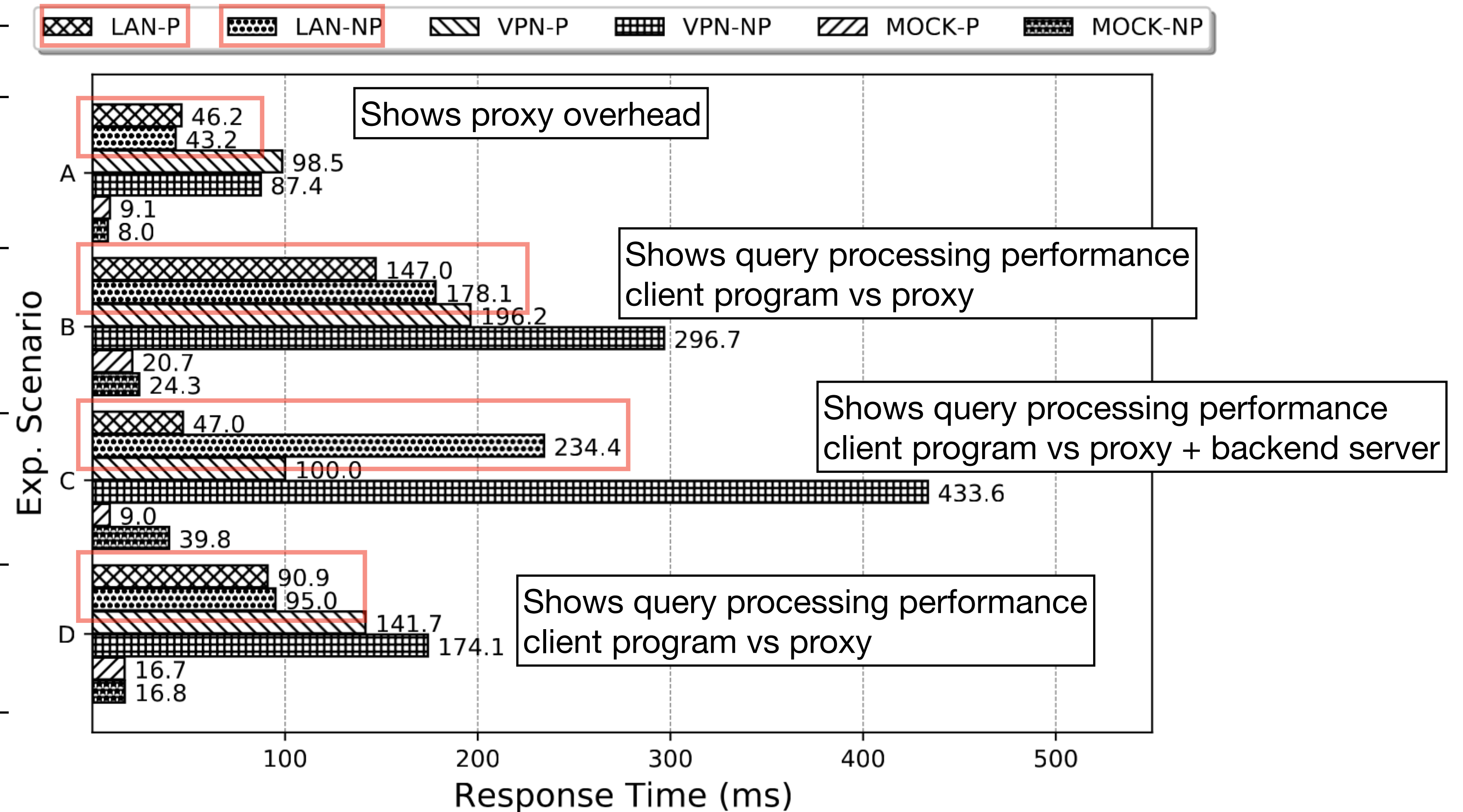
Preliminary Experimental Result

Through Proxy	Bypass Proxy
No query	No query
Generates two subrequests	Generates two subrequests
0 subrequests (delegates to Redfish service)	four subrequests
Generates one subrequest	Generates one subrequest



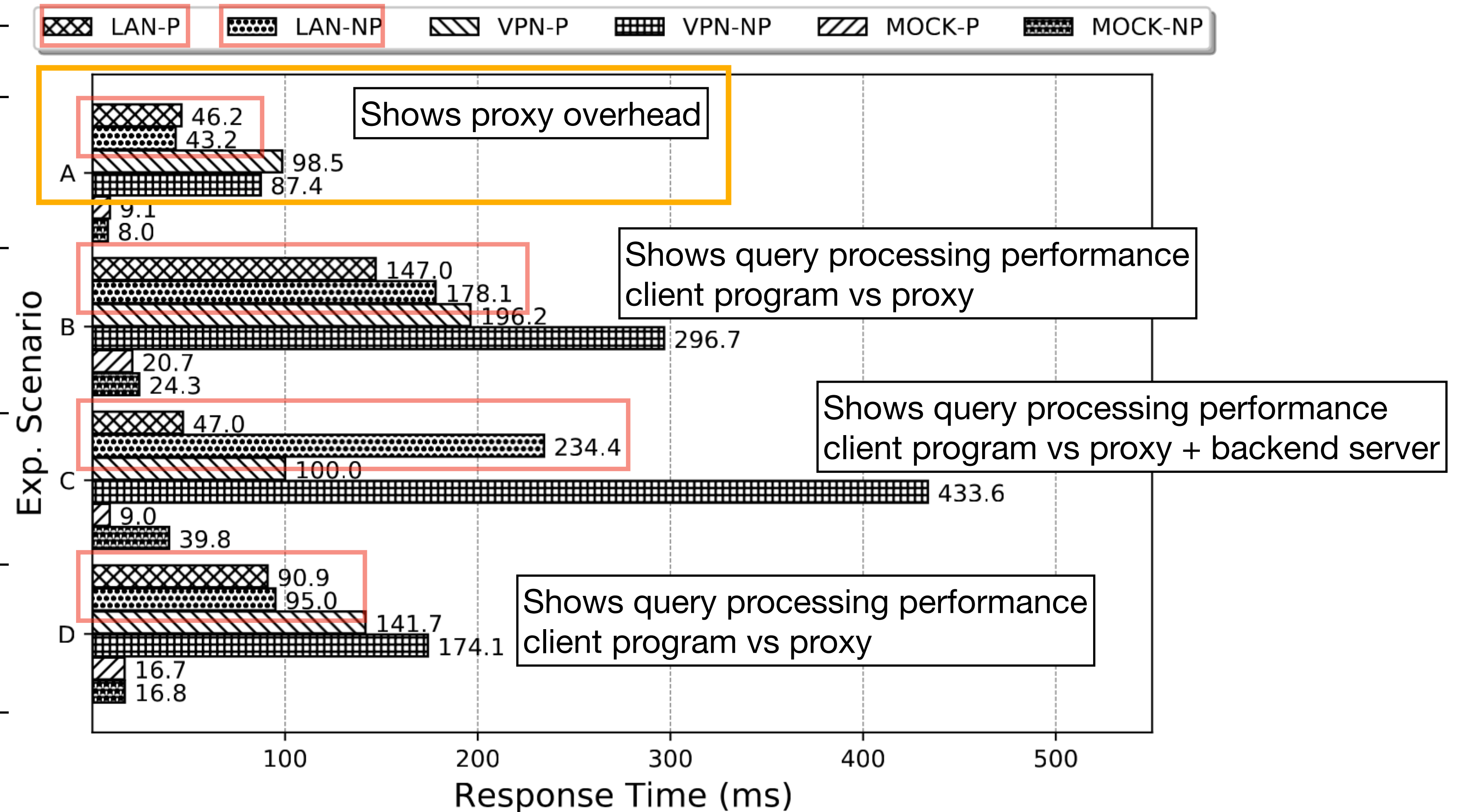
Preliminary Experimental Result

Through Proxy	Bypass Proxy
No query	No query
Generates two subrequests	Generates two subrequests
0 subrequests (delegates to Redfish service)	four subrequests
Generates one subrequest	Generates one subrequest



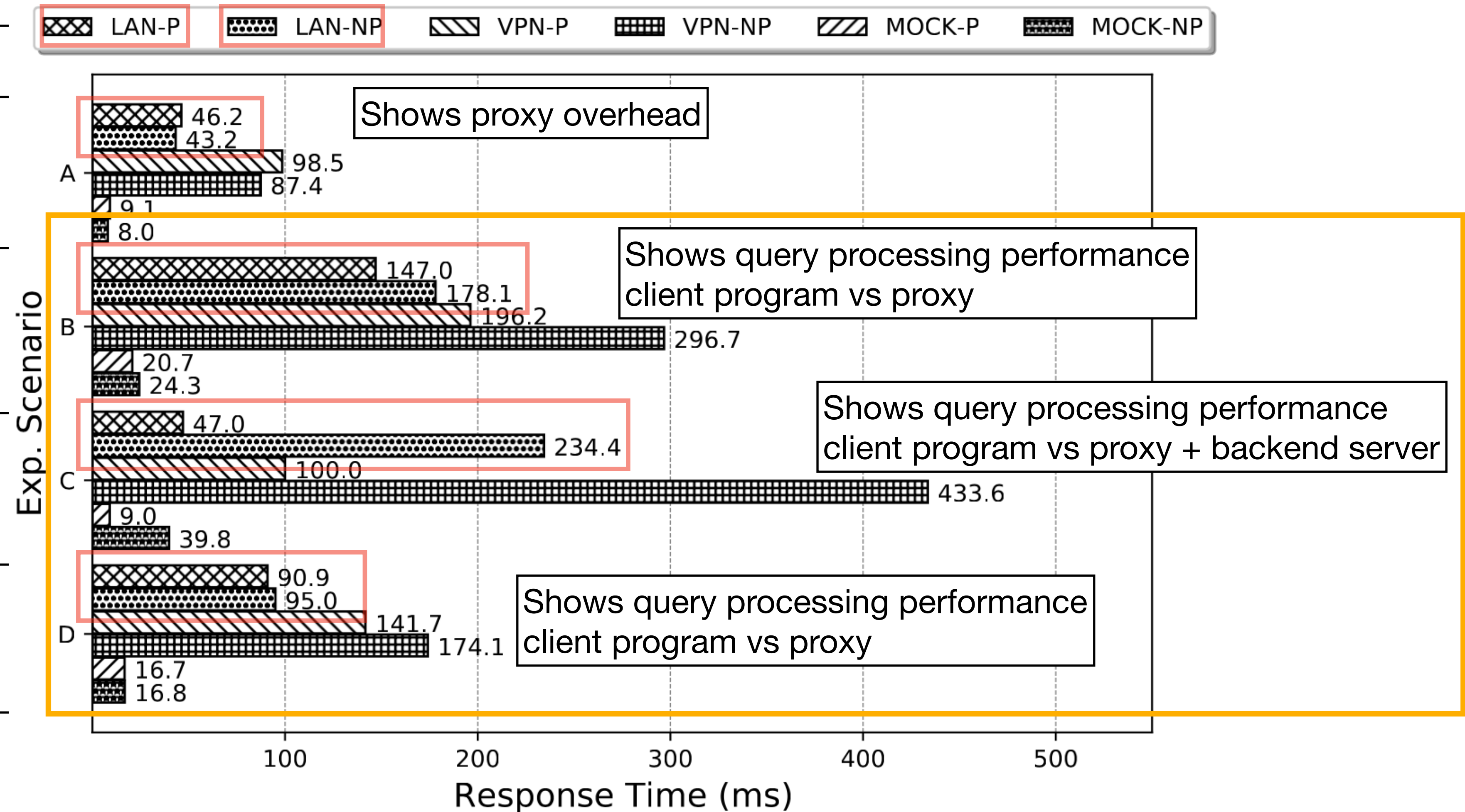
Preliminary Experimental Result

Through Proxy	Bypass Proxy
No query	No query
Generates two subrequests	Generates two subrequests
0 subrequests (delegates to Redfish service)	four subrequests
Generates one subrequest	Generates one subrequest



Preliminary Experimental Result

Through Proxy	Bypass Proxy
No query	No query
Generates two subrequests	Generates two subrequests
0 subrequests (delegates to Redfish service)	four subrequests
Generates one subrequest	Generates one subrequest



Conclusion

- Preliminary evaluation demonstrated system performance is improved in terms of query response time
- Redfish services can easily support the query functions recommended in a specification document
- We plan to use caching for performance improvements
- We plan to continue developing to support all query features specified by Redfish, and tests in various environments