UDK 004.75

HOW TO APPLY MODELING AND OPTIMIZATION TO SELECT THE APPROPRIATE CLOUD PLATFORM

B. Zibitsker† *BEZNext*

A. Lupersolsky *BEZNext*

BEZNext, Chicago, IL
E-mail: bzibitsker@beznext.com, alupersolsky@beznext.com

Abstract. Organizations want to take advantage of the flexibility and scalability of Cloud platforms. By migrating to the Cloud they hope to develop and implement new applications faster with lower cost. Amazon AWS, Microsoft Azure, Google, IBM, Oracle and other Cloud providers support different DBMS like Snowflake, Redshift, Teradata Vantage, etc. These platforms' different architecture, mechanism of allocation and management of resources, and sophistication of DBMS optimizers affect performance, scalability and cost. As a result, the response time, CPU service time and the number of I/Os for the same query accessing the similar table in the Cloud could be significantly different than On Prem.

In order to select the appropriate Cloud platform, we use modeling and optimization. First, we perform a Workload Characterization for On Prem Data Warehouse. Each Data Warehouse workload represents a specific line of business and includes activity of many users generating concurrently simple and complex queries accessing data from different tables. Each workload has different demand for resources and different response time and throughput Service Level Goals (SLG).

In this paper we will review results of the workload characterization for On Prem Data Warehouse environment.

Secondly, we collect measurement data for standard TPC-DS benchmark tests performed in AWS Vantage, Redshift and Snowflake Cloud platforms for different sizes of the data sets and different number of concurrent users.

During third step we use the results of the workload characterization and measurement data collected during the benchmark to modify BEZNext On Prem closed queueing network model to model individual Clouds.

And finally, during the fourth step we use the model to consider differences in concurrency, priorities and resource allocation to different workloads. BEZNext Capacity Planning optimization algorithms incorporate gradient search mechanism to find the AWS instance type and minimum number of instances which will be required to meet SLGs for each of the workloads. Publicly available information about the cost of the different AWS instances, storage and DBMS software is used to predict the cost of supporting workloads in the Cloud month by month during next 12 months.

Keywords: Cloud Platform, Service Level Goals, Workload Characterization, Workload Forecasting, Seasonality Determination, Benchmarking, Modeling, Optimization.

Introduction.

Organizations planning to move On Prem workloads to Cloud are looking for Cloud platform that will be able to continuously satisfy SLGs of the individual workloads (response time and throughput) with the lowest cost. Perpetual change in demand for resources caused by growth in the number of users, volume of data and implementation of new applications increase the contention for resources and cause unstable performance. Cloud elasticity addresses these problems but requires proactive actions to control performance and cost.

Every Cloud platform has a different architecture, elasticity implementation, workload management and DBMS options affecting the workloads' scalability and performance. For example, Teradata Vantage architecture [11] shown on Figure 1 uses sophisticated optimizer and DBMS management options. Database and query tuning can provide significant benefits. Change of Vantage's rules setting workloads' priorities and concurrency is used to control resource allocation and management. On the other hand, Vantage autoscaling is limited and it is not as simple to use as in Snowflake.

Figure *1.* – Teradata Vantage Architecture

Snowflake architecture [9,10] shown on Figure 2 has limited use of Materialized Views for complex queries, but it provides Virtual Warehouses to isolate workloads and reduce contention for resources. Snowflake automatically scales out by adding a cluster of EC2 instances when the number of queries to be processed exceeds a predefined level (8-32). It also scales up by changing its Virtual Warehouse size from X-Small to 4X-Large.

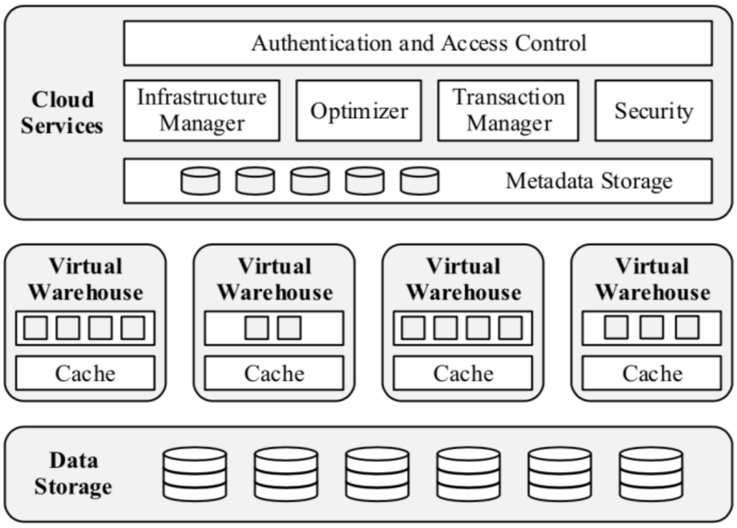

Figure *2.* – Overview of Snowflake Architecture

AWS Redshift [5,6] architecture shown on Figure 3 allows customers to select from variety of instance types but has limited concurrency management options, and there is a contention for resources between all workloads.

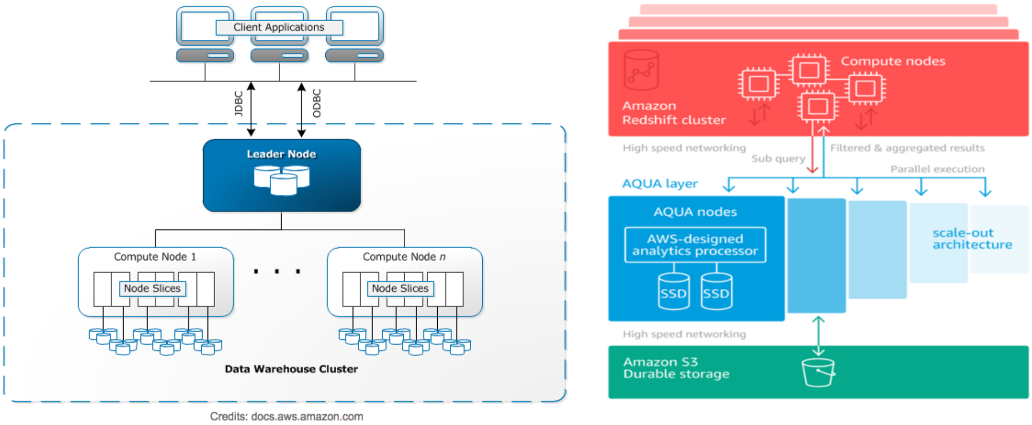

Figure 3. – Redshift Architecture

New release of Redshift based on ra3 instance types incorporates AQUA layer that accelerates queries execution by running data intensive tasks such as filtering and aggregation, compression and encryption closer to the storage layer.

Preparation to migration to the Cloud requires significant efforts to modify applications/SQL and database design.

Another challenge is to estimate how many instances will be required to meet SLGs (for example, if ETL workloads will be able to finish data load on time).

Many organizations run Proof of Concepts projects and execute representative queries against a subset of data because it is too costly to move all production data and execute production workloads in different Clouds. Typically, it is difficult to extrapolate results of the benchmark tests and compare scalability, performance and cost of configurations which will be required to host constantly changing and growing On Prem production workloads. Each workload includes a mix of DML queries generated by different numbers of concurrent users accessing databases with the various level of parallelism depending on their size and structure.

The “trial and error” approach during selection of the Cloud is too expensive. The Cloud cost over time could be very different from what was expected. In this paper we will review a methodology and use case illustrating how BEZNext modeling and performance optimization software is used to determine the minimum configuration required in each Cloud to meet SLGs for each of On Prem Data Warehouse workloads every shift of the day during next 12 months and to predict the corresponding cost.

Methodology. The presented methodology is based on use of modeling and optimization of On Prem environment and each Cloud option.

Measurement data continuously collected On Prem in the production environment are aggregated into business workloads. Workload Characterization is used to generate Performance, Resource Utilization and Data Usage profiles for each production workload, and to determine the seasonality of service demands of each workload.

Measurement data collected during standard TPC-DS benchmark tests against different data sets sizes with different numbers of concurrent users are used to determine how different Cloud Architectures change the CPU service time, and KB per query. We also analyze the differences in Cloud scalability during processing of the benchmark queries.

Results of the benchmark tests are used then to modify CPU Service time and #I/O operations for each business workload in BEZNext Queueing Network Models to model their performance in different Clouds. BEZNext optimization algorithms are used to find the minimum configuration required to meet SLGs for each workload on each of the optional Cloud platforms. We predict the impact of the workload and volume of data growth and periodical changes of workload parameters on response time and throughput for each workload On Prem and on each Cloud platform. Instance types available for each Cloud and results of the workload forecasting are used as input to the models and optimization algorithms.

One of the unique solutions presented in this paper is how optimization algorithms are applied and used to determine the minimum configuration which will be required to meet SLGs for each workload for every shift of the day during next 12 months. Predicting the minimum configuration makes it possible to estimate the cost of running production Data Warehouse workloads on different Cloud platforms and justify the final Cloud selection.

Use case

The organization is evaluating moving Teradata Data Warehouse workloads to Vantage, Redshift or Snowflake Cloud platforms.

Current Hardware Configuration On Prem

Data Warehouse is based on Teradata server with the following configuration:

Massively Parallel Processing Architecture (MPP) with 49 processing nodes:

6800 nodes, 56 CPUs per node, 512GB memory

DiskArray1: with RAID1, 90, SSD, disk capacity 1,600GB, transfer rate 450 MBps, average seek time 0.1 ms, 1017 disks.

DiskArray2: RAID1, HDD, 600GB disks with transfer rate 130 MBps, average seek time 3.5 ms, rotational speed 10,000 rpm, 3159 disks

OS SLES 11, DBMS Teradata

Interconnect Bandwidth 11200 MBps

Hardware Configurations used for Benchmark Tests

Teradata: 6700 56 CPU 512GB RAM 48 SSD 84 HDD

Vantage: AWS m4.4xlarge 16 vCPU 64GB RAM 25 SSD

Redshift: AWS dc2.large 2 vCPU 16GB RAM 1 SSD

Snowflake: AWS c5d.2xlarge 8 vCPU 16GB RAM 1 SSD

Data Collection

During this project we performed TPC-DS benchmark tests and collected performance measurement data On Prem and on each of the Cloud platforms as it shown on Figure 4 including the average values of CPU Service Time and KB I/O per query, elapsed time and execution time grouped by dataset size (1GB, 10GB and 100GB) and by the number of concurrent users (10, 20, 30, 40 and 50).

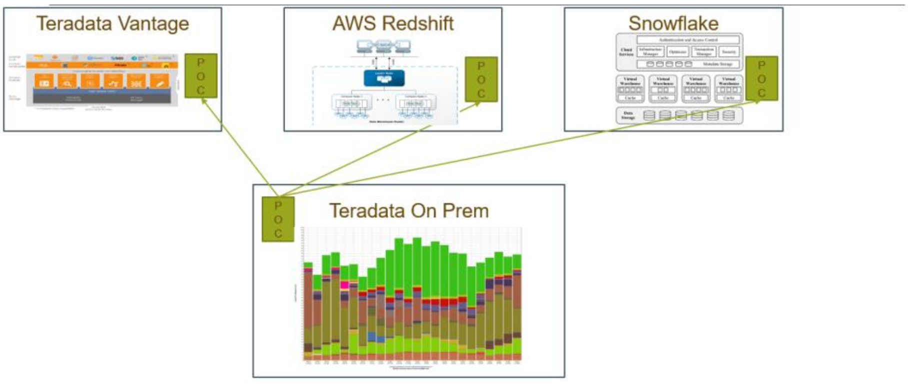

Figure *4.* – Data Collection and Workload Characterization On Prem and each of the Cloud platform

We used BEZNext data collection agents to collect measurement data On Prem from ResUsage, DBQL, TDWM tables every hour.

On each Cloud our agents extracted measurement data stored in system tables. A sample of measurement data collected during TPC-DS Benchmarks is shown in Table 1.

Table 1. – Measurement data collected during the benchmark tests On Prem and on each Cloud platform includes the average values of CPU Service Time and KB I/O per Query Type, Elapsed Time and Execution Time grouped by Data set size by the number (#) of concurrent users

Data size	# sessions	Query tag	Teradata Lab: Elapsed Time	Execution Time	CPU seconds	IO KB Log	IO KB Phys
1	10	TPCDS_Query1	0.478	0.471	3.315	1012167.570	457.549
		TPCDS_Query2	0.795	0.789	4.134	1479687.813	456.367
		TPCDS_Query3	0.643	0.627	2.905	1163046.709	241.172
		TPCDS_Query4	0.269	0.265	0.654	583340.421	97.321
		TPCDS_Query5	0.537	0.494	2.745	1017037.105	61.804
		TPCDS_Query6	0.260	0.257	0.479	403553.893	350.682
		TPCDS_Query7	0.402	0.397	1.807	984213.107	131.945
		TPCDS_Query9	0.469	0.464	4.062	807313.714	453.071
		TPCDS_Query10	0.387	0.383	2.565	812010.003	82.560
		TPCDS_Query11	0.383	0.378	2.329	915908.425	66.997
		TPCDS_Query12	3.293	3.276	42.576	6696576.583	4227.277
		TPCDS_Query13	9.649	9.645	44.864	4683308.876	5538.560
		TPCDS_Query14	0.243	0.240	1.239	558418.599	41.303
		TPCDS_Query15	0.232	0.228	1.184	603656.393	34.441
		TPCDS_Query16	0.512	0.506	1.420	861971.598	90.398
1	20	TPCDS_Query1	1.226	1.221	2.410	1013351.062	102.335
		TPCDS_Query2	1.964	1.961	4.340	1527549.733	149.958
		TPCDS_Query3	1.675	1.668	3.030	1216177.292	100.261
		TPCDS_Query4	0.817	0.815	0.662	553808.817	52.440
		TPCDS_Query5	1.340	1.321	2.788	1082482.365	81.904
		TPCDS_Query6	0.873	0.871	0.525	648448.892	44.935
		TPCDS_Query7	1.015	1.012	1.874	991098.822	73.764
		TPCDS_Query9	1.146	1.142	4.123	808355.986	68.555
		TPCDS_Query10	0.997	0.994	2.676	817268.297	84.367
		TPCDS_Query11	1.083	1.080	2.410	919245.200	73.825
		TPCDS_Query12	7.958	7.953	44.053	6988081.563	4350.794
		TPCDS_Query13	14.658	14.654	51.050	4711186.049	5495.266
		TPCDS_Query14	0.690	0.688	1.284	565712.032	46.385
		TPCDS_Query15	0.591	0.588	1.212	609769.886	66.830
		TPCDS_Query16	1.293	1.289	1.506	895335.099	80.736

Data size	# sessions	Query tag	Redshift: Elapsed Time	Execution Time	CPU seconds	IO KB
1	10	TPCDS_Query1	1.058	1.051	0.179	40163.487
		TPCDS_Query2	0.745	0.740	0.031	14113.020
		TPCDS_Query3	2.646	2.641	0.418	55542.104
		TPCDS_Query4	0.210	0.205	0.004	3877.031
		TPCDS_Query5	0.165	0.160	0.000	756.037
		TPCDS_Query6	0.322	0.317	0.052	6231.260
		TPCDS_Query7	0.481	0.477	0.081	33330.497
		TPCDS_Query9	0.264	0.259	0.013	6995.339
		TPCDS_Query10	0.501	0.497	0.053	13842.404
		TPCDS_Query11	0.471	0.466	0.036	15371.230
		TPCDS_Query12	2.420	2.356	0.510	90307.296
		TPCDS_Query13	0.329	0.325	0.006	4254.269
		TPCDS_Query14	0.284	0.280	0.020	10242.802
		TPCDS_Query15	0.238	0.234	0.015	13406.423
		TPCDS_Query16	0.281	0.276	0.007	5539.199
1	20	TPCDS_Query1	1.767	1.350	0.304	43377.778
		TPCDS_Query2	1.453	0.970	0.035	15334.895
		TPCDS_Query3	3.837	3.570	0.712	65400.354
		TPCDS_Query4	0.802	0.282	0.003	3164.719
		TPCDS_Query5	0.763	0.256	0.000	557.624
		TPCDS_Query6	0.911	0.431	0.011	5836.131
		TPCDS_Query7	1.563	1.125	0.095	37680.954
		TPCDS_Query9	0.809	0.359	0.017	6852.581
		TPCDS_Query10	1.131	0.648	0.048	12756.005
		TPCDS_Query11	1.105	0.613	0.034	14231.463
		TPCDS_Query12	1.767	1.132	0.584	101256.799
		TPCDS_Query13	0.955	0.458	0.006	3938.968
		TPCDS_Query14	0.867	0.370	0.019	9312.430
		TPCDS_Query15	0.797	0.327	0.013	9622.135
		TPCDS_Query16	0.861	0.379	0.006	4794.224

Data size	# sessions	Query tag	Vantage: Elapsed Time	Execution Time	CPU seconds	IO KB Log	IO KB Phys
1	10	TPCDS_Query1	1.214	1.109	3.410	751841.637	1099.587
		TPCDS_Query2	2.080	1.834	5.639	888590.285	2960.327
		TPCDS_Query3	2.008	1.398	4.385	840061.414	1513.586
		TPCDS_Query4	0.650	0.609	0.951	458856.323	133.750
		TPCDS_Query5	2.055	1.156	4.363	648102.971	88.792
		TPCDS_Query6	0.689	0.561	0.646	443493.184	149.388
		TPCDS_Query7	1.093	0.962	3.210	498045.424	1575.420
		TPCDS_Query9	1.269	1.100	4.544	564456.528	818.157
		TPCDS_Query10	1.228	1.083	3.267	496806.816	1778.714
		TPCDS_Query11	1.283	1.138	3.174	649564.565	121.647
		TPCDS_Query12	7.456	7.004	35.397	6206637.882	480798.138
		TPCDS_Query13	15.883	15.726	43.945	4206224.757	56135.606
		TPCDS_Query14	0.919	0.831	2.705	412880.952	70.381
		TPCDS_Query15	0.939	0.803	2.714	401961.209	80.461
		TPCDS_Query16	1.114	0.940	1.732	640734.365	271.040
1	20	TPCDS_Query1	2.346	2.344	3.455	752613.867	1196.607
		TPCDS_Query2	3.834	3.558	5.718	889456.550	2879.711
		TPCDS_Query3	3.583	2.930	4.443	853009.088	2565.447
		TPCDS_Query4	1.485	1.464	0.942	459615.815	100.049
		TPCDS_Query5	3.329	2.352	4.405	650756.645	69.674
		TPCDS_Query6	1.565	1.456	0.682	447241.425	127.486
		TPCDS_Query7	2.080	1.904	3.247	501150.791	1590.838
		TPCDS_Query9	3.315	2.126	4.574	566202.741	1548.474
		TPCDS_Query10	2.240	2.072	3.314	504720.322	3579.848
		TPCDS_Query11	2.448	2.283	3.450	652429.734	115.672
		TPCDS_Query12	13.772	13.278	35.772	6282617.622	503563.297
		TPCDS_Query13	25.045	24.869	44.652	4210973.359	17569.288
		TPCDS_Query14	1.752	1.652	2.727	413359.282	68.148
		TPCDS_Query15	1.711	1.586	2.739	402782.067	76.796
		TPCDS_Query16	2.169	1.969	1.764	642803.720	235.577

Data size	# sessions	Query tag	Snowflake: Elapsed Time	Execution Time	CPU seconds	IO KB
1	10	TPCDS_Query1	283.15	275.96	37.47	6,343,578,125.00
		TPCDS_Query2	141.86	137.43	11.89	2,878,515,625.00
		TPCDS_Query3	41.91	41.20	4.66	1,025,390,625.00
		TPCDS_Query4	16.57	12.83	8.22	334,960,937.50
		TPCDS_Query5	4.89	4.16	0.74	15,045,181.23
		TPCDS_Query6	187.93	183.99	28.02	2,285,156,250.00
		TPCDS_Query7	1.49	1.22	0.41	327,551.56
		TPCDS_Query9	157.96	154.86	122.80	2,314,453,125.00
		TPCDS_Query10	21.12	17.50	3.06	390,625,000.00
		TPCDS_Query11	251.61	247.89	51.76	2,216,796,875.00
		TPCDS_Query12	15.00	9.73	2.94	138,671,875.00
		TPCDS_Query13	21.74	18.54	0.91	368,164,062.50
		TPCDS_Query14	13.45	10.10	1.48	385,742,187.50
		TPCDS_Query15	17.50	15.93	7.95	560,156,250.00
		TPCDS_Query16				
1	20	TPCDS_Query1 – TPCDS_Query16				

Workload Characterization On Prem

On Prem measurement data are automatically aggregated into business workloads.

Results of the workload characterization are used for performance analysis, Anomaly and Root cause detection and seasonality determination.

An example of CPU consumption by different workloads for 24 hours is shown on Figure 5. Each shift has a different pattern of resource usage that will be used in our use case to decide when to allocate and deallocate resources to take advantage of Clouds' elasticity.

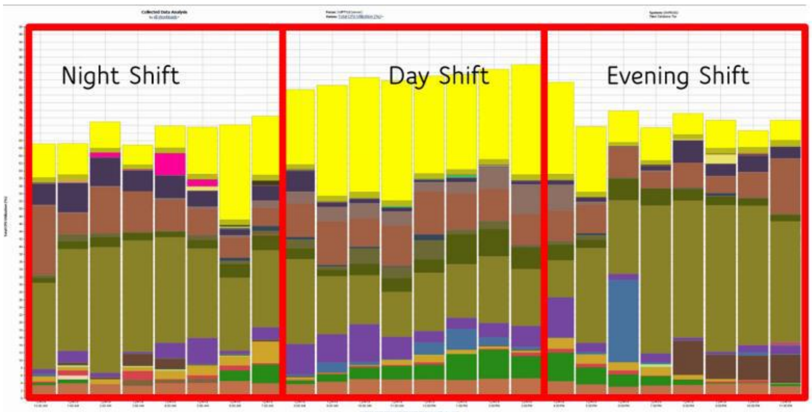

Figure 5. – Example of hourly CPU utilization for On Prem Data Warehouse workloads

Results of the workload characterization are also used to determine the Seasonal Peaks for each workload. Each Seasonal Peak is characterized by start time, duration and amplitude of peak. This information is used to ensure that sufficient resources are proactively allocated exactly when needed and deallocated after then. Automatic determination of seasonal peaks (Figure 6) is used to proactively modify rules of resource allocation and deallocation to continuously meet SLGs and lower the cost [2].

Figure 6 below shows an example of the seasonal peaks for each workload On Prem, where three workloads have peaks in resource utilization daily.

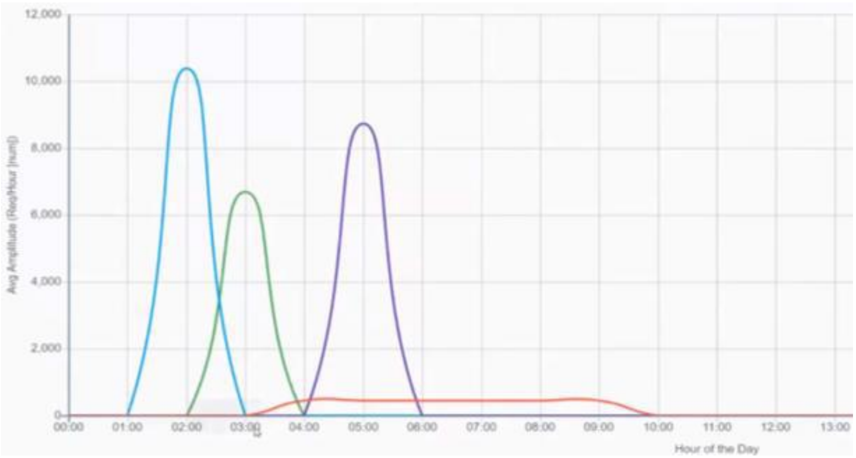

Figure *6.* – Example of determining Daily Peaks for Production Workloads

Workload Characterization Change for the Clouds

The main goal of running benchmarks on each Cloud is to compare the difference in CPU Service time and KB per query for each of the business workloads On Prem and in each of the Clouds.

For each workload we compared the TPC-DS CPU service time On Prem with CPU Service time consumed by TPC-DS queries on each of the Cloud platform and calculated the average CPU time ratio. The same for KB per query. We applied the ratios to modify service time of production workloads in the queueing network model of each Cloud.

Modeling. The goal of modeling is to predict the impact of the expected workload and volume of data growth and impact of new application implementation on performance of each workload for On Prem and for each Cloud platform. We apply Queueing Network models, ML and AI algorithms (Figure 7) to determine the minimum configuration required to meet SLGs for each workload for every hour of the Day during the next 12 months and calculate cost.

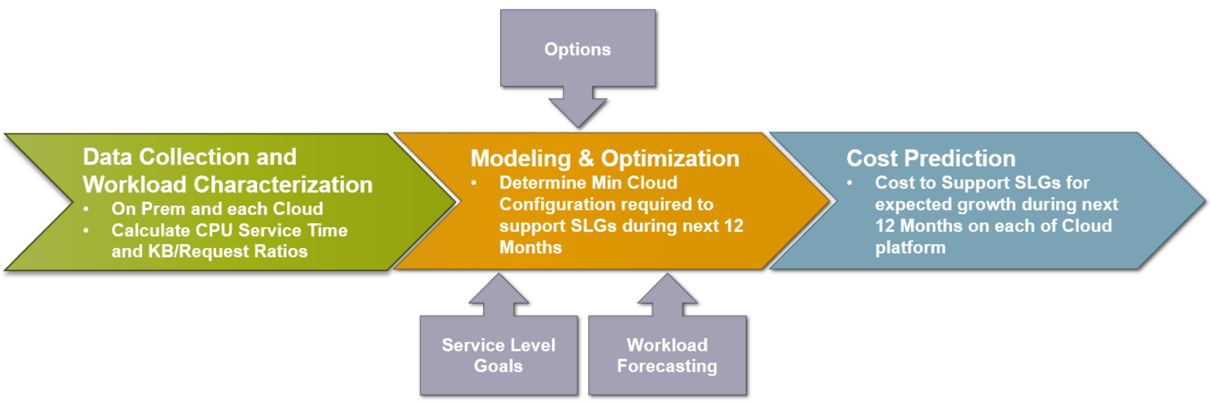

Figure *7.* – Modeling and Optimization is used to determine the minimum configuration required to meet SLGs for each workload for every hour of the Day during the next 12 months and calculate cost.

BEZNext incorporates Closed Queueing Network Models modified to be able to accurately predict not only the impact of the workload and volume of data growth, impact of changing hardware and software configuration in multi-tier, distributed, parallel processing environment, but also to predict the impact of changing Priorities, Level of concurrency and resource allocation on performance of each workload and many other "what if" questions [3]. A BEZNext Capacity Planning

Adviser based on Gradient Search optimization technology is used to find the minimum number of instances and instance types which will be required to meet SLGs for each workload every hour of the day during the next 12 months for each of the possible Cloud platforms (Figure 8).

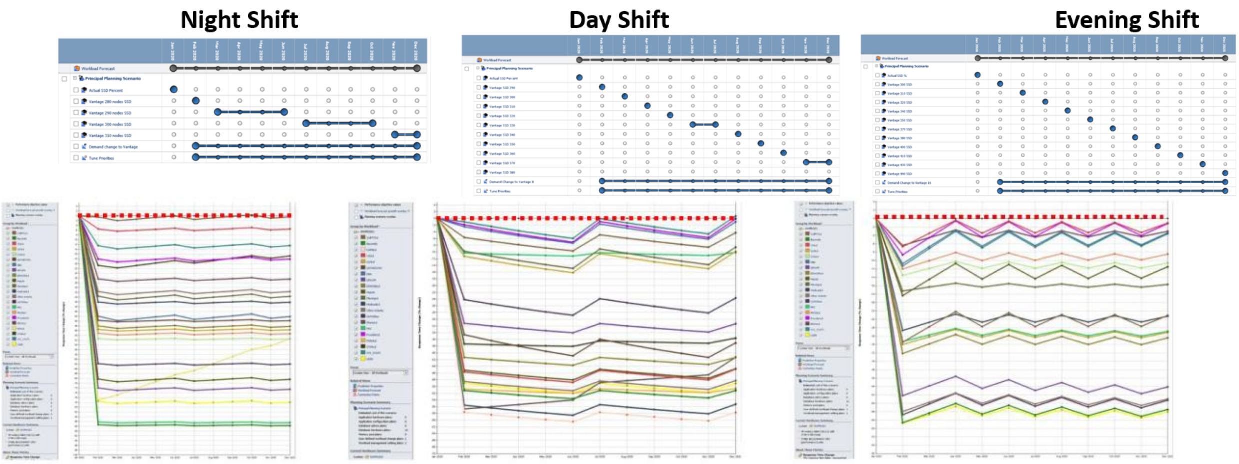

Figure *8.* – Example of predicting the minimum configuration required for Vantage to provide Response Time better than current level On Prem during next 12 months for night, day and evening shifts

Minimum configuration determined during modeling and publicly available information about cost of instances is used to predict monthly and yearly cost of running On Prem workloads on each of the optional Cloud platform.

Table 2. – Pricing information for different Instance types

Cloud Name	Instance Type	Cost per Instance / Hour	Software Advanced	Storage
Vantage	m4.16XL	$3.20	$18.86	$0.71 per 5TB per hour
Redshift	dc2.8XL	$4.80	NA	2.6TB/instance included
Redshift	ra3.16XL	$13.04	NA	$0.024 GB/month
Snowflake	2XL with 32 c5d.2Xlarge	$192/hour	NA	$23 per TB / Month
Snowflake	3XL with 64 c5d.2Xlarge	$384/hour	NA	$23 per TB / Month
Snowflake	4XL with 128 c5d.2Xlarge	$768/hour	NA	$23 per TB / Month

Conclusion. The methodology and use case illustrate how BEZNext Performance Assurance modeling and optimization software is used to evaluate AWS Redshift, Vantage and Snowflake platforms to select the best Cloud platform for Data Warehouse workloads.

We used measurement data collected during TPC-DS benchmark against different data set sizes with different number of concurrent users to estimate changes of CPU service time and KB per query for each workload in each Cloud comparing with the On Prem Data Warehouse environment.

We illustrated results of applying ML and AI algorithms for workload characterization and seasonality determination.

Queueing Network Models were used to predict the impact of the expected workload and volume of data growth on performance and resource utilization for each workload. Optimization technology was used to predict the minimum configuration of each Cloud which will be required to meet SLGs. These results were used to compare cost and select the Cloud platform with minimum cost.

References

[1] B. Zibitsker, «How to select appropriate IT infrastructure to support digital transformation», CMG Impact 2020.

[2] B. Zibitsker, «Performance Assurance for Big Data and Cloud Computing», Big Data Advanced Analytics, Minsk, 2019.

[3] J. Buzen, B. Zibitsker, «Challenges of Performance Prediction in Multi-tier Parallel Processing Environments», CMG 2006.

[4] B. Zibitsker, A. Lupersolsky, "How to Apply Modeling to Compare Options and Select the Appropriate Cloud Platform", ICPE WOSP-C 2020, Edmonton, Alberta.

[5] https://hevodata.com/blog/redshift-architecture/.

[6] https://hevodata.com/blog/amazon-redshift-pros-and-cons/.

[7] https://aws.amazon.com/redshift/.

[8] https://hevodata.com/blog/snowflake-architecture-cloud-data-warehouse/

[9] https://docs.snowflake.net/manuals/user-guide/intro-key-concepts.html.

[10] https://hevodata.com/blog/snowflake-data-warehouse/.

[11] https://www.teradata.com/Products/Software/Vantage.