PRESCRIPTIVE ANALYTICS BASED MULTI-PARAMETRIC DATABASE-COMPUTE TIER REQUISITION STACK FOR CLOUD COMPUTING

Priority

[001] This application claims priority to Indian Provisional Patent Application No. 202041018141, filed April 28, 2020, bearing Attorney Docket No. PT/2020/11688, and titled Prescriptive Analytics Based Multi-Parametric Database-Compute Tier Requisition Stack for Cloud Computing, which is incorporated by reference in its entirety.

Technical Field

[002] This disclosure relates to database-compute tier requisition via a prescriptive analytics based tier requisition stack.

Background

[003] Rapid advances in communications and storage technologies, driven by immense customer demand, have resulted in widespread adoption of cloud systems for managing large data payloads, distributed computing, and record systems. As one example, modern enterprise systems presently maintain data records many petabytes in size in the cloud. Improvements in tools for cloud resource allocation and consumption prediction will further enhance the capabilities of cloud computing systems.

Summary of Invention

[003a] According to a first aspect of the present invention, there is provided a system including: tier requisition circuitry configured to execute a tier requisition stack, the tier requisition circuitry configured to: at a data input layer of the tier requisition stack: obtain historical data including data types, the data types including processor utilization type data, operation rate type data, flush volume type data, or any combination thereof; and obtain database-compute tolerance data establishing one or more database-compute tolerances; at a coefficient layer of the tier requisition stack: access the data types via a memory resource provided by a data-staging layer of the tier requisition stack; determine a corresponding stratum from among strata for each

of the data types, where each stratum of the strata is assigned a corresponding activity coefficient; for each of the data types, obtain a corresponding type weight; and combine the corresponding activity coefficients in accord with the corresponding type weights to obtain an activity factor; at a tolerance layer of the tier requisition stack: determine an initial database-compute tier based on the activity factor, the initial database-compute tier associated with a database-compute allowance; compare the database-compute allowance to the database-compute tolerance data to determine accommodation of the one or more database-compute tolerances by the initial database-compute tier; at a time that the initial database-compute tier fails to accommodate the one or more database-compute tolerances, increase from the initial database-compute tier to an adjusted database-compute tier based on the initial database-compute tier and tolerance layer computation; at a requisition layer of the tier requisition stack: generate a provisional tier requisition token designating a request for date-compute resources at the adjusted database-compute tier; and based on a finalization directive, generate a finalized tier requisition token; and network interface circuitry configured to send the finalized tier requisition token to a host interface for control of database-compute resources.

[003b] According to a second aspect of the present invention, there is provided a method including: at a data input layer of a tier requisition stack: obtaining historical data including data types, the data types including processor utilization type data, operation rate type data, flush volume type data, or any combination thereof; and obtaining database-compute tolerance data establishing one or more database-compute tolerances; at a coefficient layer of the tier requisition stack: accessing the data types via a memory resource provided by a data-staging layer of the tier requisition stack; determining a corresponding stratum from among strata for each of the data types, where each stratum of the strata is assigned a corresponding activity coefficient; for each of the data types, obtaining a corresponding type weight; and combining the corresponding activity coefficients in accord with the corresponding type weights to obtain an activity factor; at a tolerance layer of the tier requisition stack: determining an initial database-compute tier based on the activity factor, the initial database-compute tier associated with a database-compute allowance; comparing the database-compute allowance to the database-compute tolerance data to determine accommodation of the one or more database-compute tolerances by the initial database-compute tier; at a time that the initial database-compute tier fails to

accommodate the one or more database-compute tolerances, increasing from the initial database-compute tier to an adjusted database-compute tier based on the initial database-compute tier and tolerance layer computation; at a requisition layer of the tier requisition stack: generating a provisional tier requisition token designating a request for date-compute resources at the adjusted database-compute tier; and based on a finalization directive, generating a finalized tier requisition token; and sending, via network interface circuitry, the finalized tier requisition token to a host interface for control of database-compute resources.

[003c] According to a third aspect of the present invention, there is provided a product including: machine-readable media other than a transitory signal; and instructions stored on the machine-readable media, the instructions configured to cause a machine to: at a data input layer of a tier requisition stack: obtain historical data including data types, the data types including processor utilization type data, operation rate type data, flush volume type data, or any combination thereof; and obtain database-compute tolerance data establishing one or more database-compute tolerances; at a coefficient layer of the tier requisition stack: access the data types via a memory resource provided by a data-staging layer of the tier requisition stack; determine a corresponding stratum from among strata for each of the data types, where each stratum of the strata is assigned a corresponding activity coefficient; for each of the data types, obtain a corresponding type weight; and combine the corresponding activity coefficients in accord with the corresponding type weights to obtain an activity factor; at a tolerance layer of the tier requisition stack: determine an initial database-compute tier based on the activity factor, the initial database-compute tier associated with a database-compute allowance; compare the database-compute allowance to the database-compute tolerance data to determine accommodation of the one or more database-compute tolerances by the initial database-compute tier; at a time that the initial database-compute tier fails to accommodate the one or more database-compute tolerances, increase from the initial database-compute tier; and determine an adjusted database-compute tier based on the initial database-compute tier and tolerance layer computation; at a requisition layer of the tier requisition stack: generate a provisional tier requisition token designating a request for date-compute resources at the adjusted database-compute tier; and based on a finalization directive, generate a finalized tier requisition token; and send, via network interface circuitry, the

finalized tier requisition token to a host interface for control of database-compute resources.

Brief Description of the Drawings

[004] Figure 1 shows an example multiple-layer tier requisition stack.

[005] Figure 2 shows example tier requisition stack logic.

[006] Figure 3 shows an example specific execution environment for the tier requisition stack of Figure 1.

[007] Figure 4 shows an example tier requisition (TR) -command interface.

[008] Figure 5 shows a second example TR-command interface.

[009] Figure 6 shows a third example TR-command interface.