

list to obtain the list of log files written such that the last sequence can be backed up. If new logs have been created, it can first be determined that the logs are not actively being written to by a database engine, and then the logs can be copied and the copying verified to have been performed successfully.

[0069] Another aspect that can be handled by taking advantage of various frameworks includes the management of various security aspects, such as secure keys and user credentials. Secure information such as secure keys and passwords can be stored using a secure key management system or service, such as is described in ~~co-pending~~ U.S. Patent Application No. 12/372,597, filed on February 17, 2009, issuing as U.S. Patent No. 8,245,037 on August 14, 2012, and entitled "Encryption Key Management," which is hereby incorporated herein by reference. Such a service can contain at least two versions for each credential, an 'OLD' version and the current version. A key can be rotated, for example, by uploading the new value for the key to the service, such as by using the base name for the key, and launching a workflow to propagate that key value to host managers as needed. Once that workflow is terminated successfully, such that each appropriate host has the new credential, the old version of the key can be effectively replaced with the new value. If, for any credential, the old key does not match the new key, that is an indication that a key rotation process is currently underway. A new key rotation is not started if the old key does not match the current key, as such an approach can risk losing credentials that may still be in use. A command line utility or similar interface can be used to push keys to the key management service, which can enforce this check.

[0069] A separate workflow can be defined for updating and/or rotating credentials such as host manager credentials on all host manager instances. Such an approach can utilize the same inputs as a "SendCredentials" or similar API on the each host manager, such as credential type, public key, and optional private key. In place of the credential value, however, the workflow can will accept the name of the key used to store that value in the key management service. The workflow can verify that the current value is different from the new value, and if the values are the same the workflow can terminate with an appropriate error condition. For each active host managed by the control plane, a sub-workflow can be launched that will

## CLAIMS

1. A computer-implemented method of controlling at least a configuration provisioning aspect of a relational database using a control environment, comprising:

- under control of one or more computer systems configured with executable instructions,
- receiving a Web services call from a customer to the control environment through a control interface;
- analyzing the Web services call to determine an action to be performed with respect to the configuration a provisioning aspect of at least one data instance of the a relational database, the configuration aspect relating to at least one of creating, deleting, backing up, recovery, scaling, replicating, provisioning, upgrading, patch management, failover, or capacity management of the at least one data instance, the at least one data instance residing in a data environment that is separate from the control environment, ~~which participates in implementing a relational database and~~
- storing information for the ~~determined~~ action to a job queue;
- extracting the information for the ~~determined~~ action from the job queue and generating a workflow based at least in part upon the information, the workflow comprising a series of tasks to be performed in the data environment with respect to the ~~determined~~ action;
- for each task of the workflow, sending state information for the task to a host manager for a host machine in the data environment operable to perform the task and return a response regarding performance of the task, a successful completion of each non-final task resulting in sending state information for a subsequent task of the workflow to the host~~workflow~~ manager; and
- when the a response is received from the host machine ~~indicating~~ that a final task of the workflow has been completed successfully, sending a notification to the customer that the action was successfully performed,
- wherein configuration of the at least one data instance by the customer is performed using the control interface of the control environment, and the customer is able to directly access the at least one data instance ~~relational database~~ after the action is performed using a data interface of the data environment without accessing the control environment.

2. ~~A~~ The computer-implemented method according to claim 1, further comprising:

monitoring the job queue for information for ~~any~~ additional actions to be performed with respect to the data environment.

3. ~~A~~The computer-implemented method according to claim 1, further comprising:  
providing the customer with a DNS (domain name system) address and port information enabling the customer to access the relational database using the data interface.

4. ~~A~~The computer-implemented method of controlling at least a configuration provisioning aspect of a data environment using a control environment, comprising:  
under control of one or more computer systems configured with executable instructions,  
analyzing a request received through a control interface of the control environment to determine an action to be performed with respect to the configuration a provisioning aspect of at least one data instance ~~of a relational database~~ in the data environment, the data environment being separate from the control environment;  
generating a workflow for the ~~determined~~ action, the workflow comprising at least one task to be performed in the data environment with respect to the ~~determined~~ action;  
for each task of the workflow, sending a task request to a component in the data environment operable to perform the task and return a response regarding performance of the task; and  
when ~~a~~the response is received from the component ~~host machine~~ indicating that a final task of the workflow has been completed, sending a notification through the control interface that the requested action was completed,  
wherein configuration of the at least one data instance by a user is performed exclusively via the control environment, and data in the data environment is accessible using a data interface of the data environment without accessing the control environment.

5. ~~A~~The computer-implemented method according to claim 4, wherein:  
the data is stored in at least one relational database.

6. ~~A~~The computer-implemented method according to claim 5, wherein:  
the request is a Web services call; and  
the action includes provisioning one of the at least one relational database.

7. ~~A~~The computer-implemented method according to claim 4, wherein:

the action includes at least one of creating, deleting, backing up, scaling, replicating, provisioning, upgrading, patch management, failover, or capacity management of or modifying the at least one data instance in the data environment.

8. ~~A~~The computer-implemented method according to claim 4, wherein:  
the request is a Web services call received to an externally-facing application programming interface (API) of the control environment.

9. ~~A~~The computer-implemented method according to claim 4, wherein:  
the data interface is accessible ~~an externally-facing API of the control environment accessible~~ using a DNS address for the at least one data instance.

10. ~~A~~The computer-implemented method according to claim 4, wherein:  
~~the~~ at least one ~~of the~~ tasks is selected from ~~the~~ a group consisting of creating the at least one data instance, setting a storage volume of the at least one data instance, ~~modifying the at least one data instance~~, downloading binaries for a selected engine, backup and recovery of the at least one data instance, data replication and failover, creating and managing a security group, managing user credentials, managing key rotation and credential management, and attaching to hardware to support the at least one data instance.

11. ~~A~~The computer-implemented method according to claim 4, wherein:  
the request is received to a Web services layer including components operable to perform at least one second task selected from ~~the~~ a group consisting of authenticating users based on credentials, authorizing users, throttling user requests, user input validation, and marshalling or unmarshalling requests and responses.

12. ~~A~~The computer-implemented method according to claim 4, further comprising:  
assembling the workflow using tasks selected based at least in part upon a database engine used for the at least one data instance ~~component~~ corresponding to the request.

13. ~~A~~The computer-implemented method according to claim 4, further comprising:

logging event information for the data environment to a data store in the control environment.

14. ~~A~~The computer-implemented method according to claim 4, further wherein:  
the component in the data environment is a host manager for a host machine operable to perform the tasks in the data environment, the host manager being further operable to collect and monitor health information for at least one aspect of the data environment.

15. ~~A~~The computer-implemented method of maintaining a data environment using a control environment, comprising:

under control of one or more computer systems configured with executable instructions,

provision at least one data instance in the data environment, the data environment being separate from the control environment;

periodically sending requests for status from a monitoring component of the control environment to at least one host manager in the data environment responsible for monitoring the at least one data instance; ~~of a relational database, the data environment being separate from the control environment;~~

receiving monitoring information ~~a response from the~~ at least one host manager which received one of the requests for status;

analyzing the monitoring information ~~response~~ to determine any action to be performed with respect to the data environment and generating a workflow for the ~~determined~~ action, the action involving one or more aspects of configuring~~provisioning~~ the at least one data instance and the workflow comprising at least one task to be performed in the data environment with respect to the ~~determined~~ action;

for each task of the workflow, sending state information for the task to the at least one host manager to perform the task and return a response with information regarding performance of the task, a successful completion of each non-final task resulting in sending the state information for a subsequent task of the workflow to the at least one host ~~workflow~~ manager; and

when ~~the~~ a response is received from the at least one host ~~manager~~ ~~machine~~ indicating ~~ing~~ that a final task of the workflow has been completed, storing the information for the ~~performed~~ action to a log for the data environment,

wherein configuration of the at least one data instance by a user is performed via the control environment, and data in the data environment is accessible using a data interface without accessing the control environment.

16. ~~A~~The computer-implemented method according to claim 15, wherein:

the at least one data instance corresponds to a relational database in the data environment.

17. ~~A~~The computer-implemented method according to claim 15, further comprising:

in response to analyzing the request to determine an action to be performed, storing the information for the ~~determined~~ action to a job queue, wherein the information for the action is able to be extracted from the job queue to generate the workflow for the action.

18. ~~A~~The computer-implemented method according to claim 15, further comprising:

when ~~at the response from a host manager~~ is not received within a determined period of time, sending another request for status; and

when ~~at the response from a host manager~~ is not received after a determined number of requests, determining an second action to be performed with respect to the data environment and generating a second workflow for the second~~determined~~ action.

19. ~~A~~The computer-implemented method according to claim 15, wherein: the at least one host manager is operable to collect and monitor health information for at least one aspect of the data environment for use in generating the monitoring information~~response to the request for status~~.

20. A system for controlling a data environment using a control environment, comprising:

at least one processor; and

memory including instructions that, when executed by the at least one processor, cause the system to:

analyze a request received through a control interface of the control environment to determine an action to be performed with respect to a configuration~~provisioning~~ aspect of at least one data instance ~~of a relational database in~~ at the data environment, the data environment being separate from the control environment;

generate a workflow for the ~~determined~~ action, the workflow comprising at least one task to be performed in the data environment with respect to the ~~determined~~ action;  
for each task of the workflow, send a task request to a component in the data

environment operable to perform the task and return a response regarding performance of the task; and

when ~~the~~ a response is received from the ~~component host machine~~ indicating that a final task of the workflow has been completed, send a notification through the control interface that the ~~requested~~ action was completed,

wherein configuration of the at least one data instance by a user is performed via the control environment, and data in the data environment is accessible using a data interface of the data environment without accessing the control environment.

21. ~~A~~The system according to claim 20, further comprising:  
an externally-facing API in the control environment operable to receive the request, wherein the request is a Web services call.

22. ~~A~~The system according to claim 20, wherein:  
the data interface is ~~an externally-facing API of the control environment~~  
accessible using a DNS address for one or more designated data instances that include the at least one data instance.

23. ~~A~~The system according to claim 20, further comprising:  
a Web services layer in the control environment operable to receive the request and perform at least one second task selected from ~~the~~ a group consisting of authenticating users based on credentials, authorizing users, throttling user requests, user input validation, and marshalling or unmarshalling requests and responses.

24. ~~A~~The system according to claim 20, further comprising:  
a workflow component operable to assemble the workflow using tasks selected based at least in part upon a database engine used for the at least one data instance~~component~~ corresponding to the request.

25. A computer program product embedded in a non-transitory computer-readable storage medium and including instructions that, when executed by at least one computing device, cause the at least one computing device to:

analyze a request received through a control interface of a control environment to determine an action to be performed with respect to a configuration provisioning aspect of at least one data instance of a relational database in a data environment, the data environment being separate from the control environment;

generate a workflow for the ~~determined~~ action, the workflow comprising at least one task to be performed in the data environment with respect to the ~~determined~~ action;

for each task of the workflow, send a request to a component in the data environment operable to perform the task and return a response regarding performance of the task; and

when ~~at the~~ response is received from the component ~~host machine~~ indicating that a final task of the workflow has been completed, send a notification through the control interface that the ~~requested~~ action was completed,

wherein configuration of the at least one data instance by a user is performed via the control environment, and data in the data environment is accessible using a data interface of the data environment without accessing the control environment.

26. ~~A~~The computer program product according to claim 25, further including instructions that, when executed by at least one computing device, cause the at least one computing device to:

provide an externally-facing API in the control environment operable to receive the request.

27. ~~A~~The computer program product according to claim 25, further including instructions that, when executed by at least one computing device, cause the at least one computing device to:

provide a Web services layer in the control environment operable to receive the request and perform at least one second task selected from ~~the~~ a group consisting of authenticating users based on credentials, authorizing users, throttling user requests, user input validation, and marshalling or unmarshalling requests and responses.

28. ~~A~~The computer program product according to claim 25, further including instructions that, when executed by at least one computing device, cause the at least one computing device to:

provide a workflow component operable to assemble the workflow using tasks selected based at least in part upon a database engine used for the at least one data instance~~component~~ corresponding to the request.