

What is claimed is:

1. A data processing system operative in cooperation with a host compute system, the data processing system comprising:
 - a router module comprising a host interface and a node interface, wherein the router module is communicatively coupled to the host compute system via the host interface; and
 - a compute node comprising:
 - a communications link, wherein the compute node is communicatively coupled to the node interface at the router module via the communications link;
 - a data store comprising records associated with a data processing operation;
 - a programmable logic component to execute the data processing operation in cooperation with the data store;
 - a node memory comprising data and instructions to support operation of the programmable logic component in connection with the data processing operation and to facilitate data communications via the communications link;
 - a data mover component to facilitate data communications between the programmable logic component and the node memory; and
 - a storage interface component to facilitate data communications between the programmable logic component and the data store;wherein the storage interface component utilizes a plurality of communications channels to transfer data between the programmable logic component and the data store.
2. The data processing system of claim 1 wherein the communications link employs a serial communications protocol.
3. The data processing system of claim 2 wherein the compute node is communicatively coupled to an additional compute node in series via the communications link.
4. The data processing system of claim 1 wherein the data store comprises an Electronically Erasable Programmable Read Only Memory (EEPROM) device.
5. The data processing system of claim 4 wherein the storage interface component employs and Open NAND Flash Interface (ONFI) protocol.

6. The data processing system of claim 1 wherein the programmable logic component comprises a field programmable gate array (FPGA).
7. The data processing system of claim 1 further comprising a processor to manage operation of the data mover component.
8. The data processing system of claim 7 wherein the data mover component comprises a device controller.
9. The data processing system of claim 1 wherein the host compute system executes instructions to influence operation of the compute node.
10. The data processing system of claim 1 wherein the router module and the compute node are integrated into an interface card communicatively coupled to the host compute system.
11. A data processing system operative in cooperation with a host compute system, the data processing system comprising:

a router module comprising a host interface and a node interface, wherein the router module is communicatively coupled to the host compute system via the host interface;
and

a compute node comprising:

a communications link, wherein the compute node is communicatively coupled to the node interface at the router module via the communications link;

a data store comprising datasets associated with a data processing operation;

a programmable logic component to execute the data processing operation in cooperation with the data store;

a node memory comprising data and instructions to support operation of the programmable logic component in connection with the data processing operation and to facilitate data communications via the communications link;

a data mover component to facilitate data communications between the programmable logic component and the node memory; and

a storage interface component to facilitate data communications between the programmable logic component and the data store;

wherein the programmable logic component executes a set of instructions to reformat a block of data comprising original records, each original record comprising a plurality of fields having different field types, such that a reformatted block of data comprises new records, each new record comprising a plurality of fields of a single field type from the original records; and
wherein the programmable logic component causes the new records to be stored as the datasets in the data store.

12. The data processing system of claim 11 wherein the compute node is communicatively coupled to an additional compute node in series via the communications link.

13. The data processing system of claim 11 wherein the data store comprises an Electronically Erasable Programmable Read Only Memory (EEPROM) device.

14. The data processing system of claim 11 wherein the programmable logic component comprises a field programmable gate array (FPGA).

15. The data processing system of claim 11 wherein the router module and the compute node are integrated into an interface card communicatively coupled to the host compute system.

16. The data processing system of claim 11 wherein the set of instructions causes the programmable logic component to execute a method comprising:

receiving a block of data, the block of data comprising a number, y , of original records wherein each original record comprises a number, x , of original fields;

for $i = 1$ to x :

defining a respective new data record, each respective new record comprising y new fields, wherein each respective new field corresponds to a respective i^{th} original field from a respective original record;

reformatting the block of data responsive to the defining such that a reformatted block of data comprises the new data records, each comprising a single field type;

associating each of the new data records with an index operative to identify a particular new data record in the reformatted block of data; and

writing the reformatted block of data as the datasets in the data store.

17. A method of executing an algorithm in a distributed processing compute environment; the method comprising:

providing an initiating compute node comprising a data store and a first node memory;
serially coupling one or more additional compute nodes to the initiating compute node to establish an execution pipeline comprising the initiating compute node and the one or more additional compute nodes, each respective one of the one or more additional compute nodes comprising a respective second node memory;
initiating the execution pipeline by loading a bit stream comprising an instruction set related to execution of the algorithm into a respective programmable logic space of each respective compute node in the execution pipeline;
executing a first operation associated with the algorithm at the initiating node using information in the bit stream and data retrieved from the data store, and passing results of the first operation to a first additional compute node in the execution pipeline;
wherein the first additional compute node and each subsequent additional compute node in the execution pipeline executes a respective additional operation associated with the algorithm using information in the bit stream and results of a respective preceding operation, and passes respective additional results of the respective additional operation to a next additional compute node in the execution pipeline; and
terminating execution of the algorithm at a management node that receives the additional results from a terminal additional compute node in the execution pipeline, the terminating comprising executing a final operation to produce a final result of the algorithm.

18. The method of claim 17 further comprising, responsive to the terminating, transmitting the final result of the algorithm to a host compute system.

19. The method of claim 17 further comprising, prior to initiating the execution pipeline, reformatting a block of data comprising original records to be used in connection with execution of the algorithm such that a reformatted block of data comprises new records, each new record comprising a plurality of fields of a single field type from the original records.

20. The method of claim 19 wherein the reformatting comprises:

receiving the block of data, the block of data comprising a number, y , of original records wherein each original record comprises a number, x , of original fields;
for $i = 1$ to x :

defining a respective new data record, each respective new record comprising y new fields, wherein each respective new field corresponds to a respective i^{th} original field from a respective original record;

reformatting the block of data responsive to the defining such that a reformatted block of data comprises the new data records, each comprising a single field type;

associating each of the new data records with an index operative to identify a particular new data record in the reformatted block of data; and

writing the reformatted block of data as the datasets in the data store.



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/729,727	12/30/2019	Robert Bismuth	FERM0003U.464609	8871
27799	7590	06/08/2021	EXAMINER	
COZEN O'CONNOR			SHIN, CHRISTOPHER B	
3WTC, 175 Greenwich Street				
55th Floor				
NEW YORK, NY 10007				
			ART UNIT	PAPER NUMBER
			2181	
			NOTIFICATION DATE	DELIVERY MODE
			06/08/2021	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patentdocket@cozen.com
patentsecretary@cozen.com
patentsorter@cozen.com

DETAILED ACTION

Notice of Pre-AIA or AIA Status

1. The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.
2. Claims 1-20 are presented and pending in the application. After careful consideration and search of the recited claimed invention, the examiner agrees with the claim rejections of the International Search Report for PCT/US19/68892. The examiner notes that broadest possible reasonable interpretations are applied by the examiner to the claimed invention.

Claim Rejections - 35 USC § 102

3. In the event the determination of the status of the application as subject to AIA 35 U.S.C. 102 and 103 (or as subject to pre-AIA 35 U.S.C. 102 and 103) is incorrect, any correction of the statutory basis for the rejection will not be considered a new ground of rejection if the prior art relied upon, and the rationale supporting the rejection, would be the same under either status.

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a)(1) the claimed invention was patented, described in a printed publication, or in public use, on sale, or otherwise available to the public before the effective filing date of the claimed invention.

4. Claims 1-20 are rejected under 35 U.S.C. 102(a)(1) as being anticipated by Whitehead et al. (US 2015/0301900 A1).

a. The examiner relies on the entire teachings of the Whitehead reference for this rejection; the applicant should carefully consider the entire teachings of the Whitehead reference to better understand the examiner's position & the reasonable broadest possible interpretations applied by the examiner.

b. The Whitehead reference teaches the all of the functionally equivalent limitations, when the reasonable broad interpretations are applied to the claimed invention, of the recited claimed as follows:

<u>Claims 1-20</u>	<u>Whitehead reference teachings</u>
--------------------	--------------------------------------

1. A data processing system operative in cooperation with a host compute system, the data /processing system comprising: a router module comprising a host interface and a node interface, wherein the router module is communicatively coupled to the host computer system via the host interface; and a

- Teachings of para [0060] Each data center has routers 132 that communicate to load proxy machines 134 to distribute incoming requests to a plurality of application servers, para [0062] Client software 110 communicates with the router 132. For management, the client software 110 also communicates with a system management platform (SMP) 120, which is controlled over a web interface 140.

a compute node comprising: a communications link, wherein the compute node is communicatively coupled to the node interface at the router module via the communications link;

- Teachings of para [0050] Back up data from the client's computer is provided to storage directors 30 that send the information to a data storage array 32, para [0062] Client software 110 communicates with the router 132. For management, the client software 110 also communicates with a system management platform (SMP) 120, which is controlled over a web interface 140, para [0351] A programmable control device may be a single computer processor, a plurality of computer processors coupled by a communications link

a data store comprising records associated with a data processing operation;

- Teachings of para [0072] the backup systems described herein will operate with data storage devices that store different

formats of data and different types of files. For example, the data storage devices may store data files, executable files, registry information, database structures and other conventional data formats and data types

a programmable logic component to execute the data processing operation in cooperation with the data store;

- Teachings of para [0072] the backup systems described herein will operate with data storage devices that store different formats of data and different types of files. For example, the data storage devices may store data files, executable files, registry information, database structures and other conventional data formats and data types; para [0351] the functional modules of the remaining figures may be embodied as a series of instructions organized into one or more computer programs which are executable by a programmable control device. A programmable control device may be a single computer processor, a plurality of computer processors coupled by a communications link

a node memory comprising data and instructions to support operation of the programmable logic component in connection with the data processing operation and to facilitate data communications via the communications link;

- Teachings of para [0071] the depicted servers may be conventional files servers of type commonly employed with computer networks and can comprise a PC compatible work station running the windows NT, UNIX, Linux, or other operating system and having sufficient random access memory and persistent memory to operate efficiently as a server systems; para [0211] At the receiving end the WebDAV server intelligently decrypts, expands the patch sets, and stamps them down onto the file system (67S). This results in the transfer efficiency of an incremental backup, but generates a full backup with snap shots going back in time for version history

a data mover component to facilitate data communications between the programmable logic component and the node memory; and

- Teachings of para [0130] The WebDAV protocol makes the Web a readable and writable medium. It provides a framework for users to create, change and move documents on a server; typically a web server or web share; para [0314], the data mover can transfer data over the internet at the maximum possible speed. The system can handle sub-file change detection. Many programs, databases in particular, continuously update existing files

a storage interface component to facilitate data communications between the programmable logic component and the data store;

- Teachings of para [0048] The software 10 is able to perform network file reads and writes, change detection, snapshots,

database serialization and many more backup and restore functions and is remote controlled by the a System Management Platform; para [0100] One embodiment of the UI component 420 provides a user interface for local users of the host computer. The UI can communicate to the service 410 via a shared pipe and can send commands to the service and monitor progress;

wherein the storage interface component utilizes a plurality of communications channels to transfer data between the programmable logic component and the data store.

- para [0076] Because of the ubiquity of IP networks, iSCSI can be used to transmit data over local area networks (LANs), wide area networks (WANs), or the Internet and can enable location-independent data storage and retrieval; para [0078] iSCSI is one of two main approaches to storage data transmission over IP networks; the other method, Fibre Channel over IP (FCIP), translates Fibre Channel control codes and data into IP packets for transmission between geographically distant Fibre Channel SANs

2. The data processing system of claim 1 wherein the communications link employs a serial communications protocol.

- Teachings of para [0064], [0101]

3. The data processing system of claim 2 wherein the compute node is communicatively coupled to an additional compute node in series via the communications link.

- Teachings of para [0064], [0099]-[0100], [0351]

4. The data processing system of claim 1 wherein the data store comprises an Electronically Erasable Programmable Read Only Memory (EEPROM) device.

- Teachings of para [0351]

5. The data processing system of claim 4 wherein the storage interface component employs an Open NAND Flash Interface (ONFI) protocol.

- Teachings of para [0351] where it teaches “communications link” with “flash device” which are functionally equivalent teachings of the claimed invention

6. The data processing system of claim 1 wherein the programmable logic component comprises a field programmable gate array (FPGA).

- Teachings of para [0351]

7. The data processing system of claim 1 further comprising a processor to manage operation of the data mover component.

- Teachings of para [0314], [0332]

8. The data processing system of claim 7 wherein the data mover component comprises a device controller.

- Teachings of para [0073], [0077], [0314]

9. The data processing system of claim 1 wherein the host compute system executes instructions to influence operation of the compute node.

- Teachings of para [0060], [0100], [0279], [0314]

10. The data processing system of claim 1 wherein the router module and the compute node are integrated into an interface card communicatively coupled to the host compute system.

- Teachings of para [0062], [0071], [0075]

11. A data processing system operative in cooperation with a host compute system, the data processing system comprising: a router module comprising a host interface and a node interface, wherein the router module is communicatively coupled to the host compute system via the host interface; and

- The teachings of para [0060] Each data center has routers 132 that communicate to load proxy machines 134 to distribute incoming requests to a plurality of application servers, para [0062] Client software 110 communicates with the router 132. For management, the client software 110 also communicates with a system management platform (SMP) 120, which is controlled over a web interface 140

a compute node comprising: a communications link, wherein the compute node is communicatively coupled to the node interface at the router module via the communications link;

- The teachings of para [0050] Back up data from the client's computer is provided to storage directors 30 that send the information to a data storage array 32, para [0062] Client software 110 communicates with the router 132. For management, the client software 110 also communicates with a system management platform (SMP) 120, which is controlled over a web interface 140, para [0351] A programmable control device may be a single computer processor, a plurality of computer processors coupled by a communications link

a data store comprising datasets associated with a data processing operation

- The teachings of para [0072] the backup systems described herein will operate with data storage devices that store different formats of data and different types of files. For example, the data storage devices may store data files, executable files, registry information, database structures and other conventional data formats and data types

a programmable logic component to execute the data processing operation in cooperation with the data store;

- The teachings of para [0072] the backup systems described herein will operate with data storage devices that store different formats of data and different types of files. For example, the data storage devices may store data files, executable files, registry information, database structures and other conventional data formats and data types; para [0351] the functional modules of the remaining figures may be embodied as a series of instructions organized into one or more computer programs which are executable by a programmable control device. A programmable control device may be a single computer processor, a plurality of computer processors coupled by a communications link

a node memory comprising data and instructions to support operation of the programmable logic component in connection with the data processing operation and to facilitate data communications via the communications link;

- The teachings of para [0071] the depicted servers may be conventional files servers of type commonly employed with computer networks and can comprise a PC compatible work station running the windows NT, UNIX, Linux, or other operating system and having sufficient random access memory and persistent memory to operate efficiently as a server systems; para [0211] At the receiving end the WebDAV server intelligently decrypts, expands the patch sets, and stamps them down onto the file system (67S). This results in the transfer efficiency of an incremental backup, but generates a full backup with snap shots going back in time for version history

a data mover component to facilitate data communications between the programmable logic component and the node memory; and

- The teachings of para [0130] The WebDAV protocol makes the Web a readable and writable medium. It provides a framework for users to create, change and move documents on a server; typically a web server or web share; para [0314] The data mover can transfer data over the internet at the maximum possible speed. The system can handle sub-file change detection. Many programs, databases in particular, continuously update existing files

a storage interface component to facilitate data communications between the programmable logic component and the data store;

- The teachings of para [0048] The software 10 is able to perform network file reads and writes, change detection, snapshots, database serialization and many more backup and restore functions and is remote controlled by the a System Management Platform, para [0100] One embodiment of the UI component 420 provides a user interface for local users of the host computer. The UI can communicate to the service 410 via

a shared pipe and can send commands to the service and monitor progress

wherein the programmable logic component executes a set of instructions to reformat a block of data comprising original records, each original record comprising a plurality of fields having different field types, such that a reformatted block of data comprises new records, each new record comprising a plurality of fields of a single field type from the original records; and

- The teachings of para [0148] Manifest entries may include internal or envelope fields used to validate record and/or file integrity up to that point such as hashes, para [0202] The merge sort may be performed before the backup process resumes with the old manifest file describing slave contents as they were before the backup combined with the incomplete manifest file from the incomplete backup to produce a third manifest file which is then used for input., para [0288] Each sample that is collected is packaged into a message format that includes an identifier for the volume (volume id), a timestamp and the metrics that are collected. There can be one message for each metric or a combination of one or more messages that include multiple metrics in a message

wherein the programmable logic component causes the new records to be stored as the datasets in the data store.

- The teachings of para [0073] data sent over the network to the library may employ a tape expiration technique wherein the tape holding or storing the oldest data is employed by the system for storing new data into the library. The controller to implement this process may be a software process operating on the backup server

12. The data processing system of claim 11 wherein the compute node is communicatively coupled to an additional compute node in series via the communications link.

- The teachings of para [0064], [0099]-[0100], [0351]

13. The data processing system of claim 11 wherein the data store comprises an Electronically Erasable Programmable Read Only Memory (EEPROM) device.

- The teachings of para [0351]

14. The data processing system of claim 11 wherein the programmable logic component comprises a field programmable gate array (FPGA).

- The teachings of para [0351]

15. The data processing system of claim 11 wherein the router module and the compute node are integrated into an interface card communicatively coupled to the host compute system.

- The teachings of para [0062], [0071], [0075]).

16 The data processing system of claim 11 wherein the set of instructions causes the programmable logic component to execute a method comprising: receiving a block of data, the block of data comprising a number, y , of original records wherein each original record comprises a number, x , of original fields;

- The teachings of para [0149], [0240], [0294], [0314]

for $i = 1$ to x defining a respective new data record, each respective new record comprising y new fields, wherein each respective new field corresponds to a respective original field from a respective original record;

- The teachings of para [0148]-[0149], [0197], [0314]

reformatting the block of data responsive to the defining such that a reformatted block of data comprises the new data records, each comprising a single field type;

- The teachings of para [0148]-[0150], [0288], [0314]

associating each of the new data records with an index operative to identify a particular new data record in the reformatted block of data; and

- The teachings of para [0127]-[0128], [0318], [0336])

writing the reformatted block of data as the datasets in the data store.

- The teachings of para [0113], [0127], [0240], [0294]

17. A method of executing an algorithm in a distributed processing compute environment; the method comprising: providing an initiating compute node comprising a data store and a first node memory;

- The teachings of para para [0060] Each data center has routers 132 that communicate to load proxy machines 134 to distribute incoming requests to a plurality of application servers, para [0062] Client software 110 communicates with the router 132. For management, the client software 110 also communicates with a system management platform (SMP) 120, which is controlled over a web interface 140., para [0071] can comprise a PC compatible work station running the windows NT, UNIX, Linux, or other operating system and having sufficient random access memory and persistent memory to operate efficiently as a server systems

serially coupling one or more additional compute nodes to the initiating compute node to establish an execution pipeline comprising the initiating compute node and the one or more additional compute nodes, each respective one of the one or more additional compute nodes comprising a respective second node memory;

- The teachings of para [0050] Back up data from the client's computer is provided to storage directors 30 that send the information to a data storage array 32, para [0062] Client software 110 communicates with the router 132. For management, the client software 110 also communicates with a system management platform (SMP) 120, which is controlled over a web interface 140, para [0099] the Service 410 starts a Sync process and monitors a shared pipe to communicate

status and errors back to the SMP. The Sync component 430 can also detect process deadlocks, stalled syncs or other errors and can proactively cancel the sync process when necessary., para [0351] A programmable control device may be a single computer processor, a plurality of computer processors coupled by a communications link

initiating the execution pipeline by loading a bit stream comprising an instruction set related to execution of the algorithm into a respective programmable logic space of each respective compute node in the execution pipeline

- The teachings of para [0066] evenly distributes access requests onto the engine nodes by running a load balance algorithm., para [0143] The manifest entries may also associate intermediate files with alternate data streams or file forks on the master and/or slave copy where the intermediate format does not support such associations., para [0193] Using the uploaded manifest file, all metadata (and optionally on-disk payload via hash compare) is verified and compared with the on-disk state.);

executing a first operation associated with the algorithm at the initiating node using information in the bit stream and data retrieved from the data store, and passing results of the first operation to a first additional compute node in the execution pipeline

- The teachings of para [0242] First, the manifest is used to identify locally what files have changed. This is done by traversing all the files in a subdirectory and comparing the cached status of the corresponding file on the server end.);

wherein the first additional compute node and each subsequent additional compute node in the execution pipeline executes a respective additional operation associated with the algorithm using information in the bit stream and results of a respective preceding operation, and passes respective additional results of the respective additional operation to a next additional compute node in the execution pipeline

- The teachings of para [0242] If the file is the same it is ignored. If the file is new it is marked for upload and if it has changed, it is passed to the sub-file change detector., para [0243] Next, a local sub-file signature cache is consulted to detect insertions, removals, and rearrangements of data in files, so that only the changed portion of the file is transmitted. Changes are wrapped into a container and marked for transmission); and

terminating execution of the algorithm at a management node that receives the additional results from a terminal additional compute node in the execution pipeline, the terminating comprising executing a final operation to produce a final result of the algorithm

- The teachings of para [0271] When the file is read by Client software 10 network mounts will be created to the defined data sources and the Client software 10 agent will trigger a snapshot on the NetApp appliance which is used to capture a consistent state of the NetApp file system and replicate the targeted data to the System cloud., para [0277] Once the transfer of data

is complete System creates a snapshot of the file system on the System storage end

18. The method of claim 17 further comprising, responsive to the terminating, transmitting the final result of the algorithm to a host compute system.

- The teachings of para [0211], [0245], [0251]

19. The method of claim 17 further comprising, prior to initiating the execution pipeline, reformatting a block of data comprising original records to be used in connection with execution of the algorithm such that a reformatted block of data comprises new records, each new record comprising a plurality of fields of a single field type from the original records.

- The teachings of para [0113], [0142], [0148]-[0149], [0199]

20. The method of claim 19 wherein the reformatting comprises: receiving the block of data, the block of data comprising a number, y, of original records wherein each original record comprises a number, x, of original fields;

- The teachings of para [0149], [0240], [0294], [0314]

for i = 1 to x: defining a respective new data record, each respective new record comprising y new fields, wherein each respective new field corresponds to a respective original field from a respective original record;

- The teachings of para [0148]-[0149], [0197], [0314]

reformatting the block of data responsive to the defining such that a reformatted block of data comprises the new data records, each comprising a single field type

- The teachings of para [0148]-[0150], [0288], [0314]

associating each of the new data records with an index operative to identify a particular new data record in the reformatted block of data; and

- The teachings of para [0127]-[0128], [0318], [0336]

writing the reformatted block of data as the datasets in the data store.

- The teachings of para [0113], [0127], [0240], [0294]

c. As can be seen from the above detailed teachings, the recited claimed

invention is anticipated by the functionally equivalent teachings of the Whitehead reference.

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to CHRISTOPHER B SHIN whose telephone number is (571)272-4159. The examiner can normally be reached on 8:00-4:00 PM.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at <http://www.uspto.gov/interviewpractice>.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, IDRIS N ALROBAYE can be reached on 571-270-1023. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <https://ppair-my.uspto.gov/pair/PrivatePair>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/CHRISTOPHER B SHIN/
Primary Examiner, Art Unit 2181