



Serial No. 09/703,942

10-06-04

2664  
41

Attorney Docket No.: HP10008102-1

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Pulsipher, et al. Examiner: Schultz, William C.  
 Serial No.: 09/703,942 Art Unit: 2664  
 Filed: 10/31/2000  
 For: METHOD AND SYSTEM FOR IDENTIFYING AND PROCESSING  
 CHANGES TO A NETWORK TOPOLOGY

Box Non-Fee Amendment  
 Commissioner of Patents  
 P.O. Box 1450  
 Alexandria, VA 22313-1450

**RECEIVED**  
 OCT 13 2004  
 Technology Center 2600

AMENDMENT AND REQUEST FOR RECONSIDERATION

Sir:

In response to the July 6, 2004, Office Action, Applicant hereby submits the following Amendment and Request for Reconsideration. Please amend the above-identified application as follows:

**Amendments to the Claims** are reflected in the listing of claims which begins on page 2 of this paper.

**Remarks/Arguments** begin on page 5 of this paper.

I hereby certify that the document is being deposited with the United States Postal Service as express mail post office to addressee in an envelope addressed to: Mail Stop Non-Fee Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on:

10-4-2004 ER 2649 11265 US  
 (Date of Deposit and Express Mail No.)  
Shane P. Coleman  
 (Name)  
Shane P. Coleman  
 Signature

This listing of claims will replace all prior versions, and listings, of claims in the application:

**LISTING OF CLAIMS**

1. (currently amended) In a network having interconnected nodes with data tuples that represent nodal connections, a method for mapping a network topology by identifying changes between an existing topology and a new topology, the method comprising:

creating a plurality of tuples for a topology of a network, wherein the tuples represent nodal connections of the network and wherein each of the tuples comprises a host identifier, interface information, and a port specification;

~~converting an existing topology into a list of existing tuples that represent existing nodal connections;~~

receiving new tuples that represent new nodal connections; and

comparing the list of existing tuples with the new tuples to identify changes to the topology.

2. (original) The method of claim 1, further comprising updating a topology database with a new topology.

3. (original) The method of claim 1, further comprising taking action on the changes to the topology.

4. (canceled)

5. (original) The method of claim 1, wherein the step of comparing comprises identifying duplicate tuples that appear both in the list of existing tuples and in the new tuples, and maintaining a current status of the topology for these tuples.

6. (original) The method of claim 1, wherein the step of comparing comprises identifying a swapped port condition on a connector.

7. (original) The method of claim 1, wherein the step of comparing comprises searching for a host of a new singly-heard host link tuple or a new multi-heard host link tuple in the list of existing tuples.

8. (currently amended) A system for mapping a network topology by identifying changes between an existing topology and a new topology, based on changes to data tuples that represent nodal connections comprising:

a topology database that stores an existing topology of a network using tuples, wherein each tuple includes a host identifier, interface information, and a port specification for a node in the network; and

a topology converter connected to the topology database that receives new tuples that represent new nodal connections, and compares the new tuples with the existing topology to identify changes in the network by comparing the host identifiers, the interface information, and the port specifications.

9. (currently amended) The system of claim 8, wherein the topology converter creates the tuples for the topology of the network. ~~converts the existing topology into a list of existing tuples that represent existing nodal connections.~~

10. (original) The system of claim 8, wherein the topology converter updates the topology database with a new topology based on the new tuples.

11. (original) The system of claim 8, wherein the topology converter attempts to identify swapped ports on connectors.

12. (original) The system of claim 8, wherein the topology converter identifies duplicate tuples that appear both in the list of existing tuples and in the new tuples, and maintains a current status of the topology for these tuples.

13. (original) The system of claim 8, wherein the topology converter searches for a host of a new singly-heard host link tuple or a new multi-heard host link tuple in the list of existing tuples.

14. (original) The system of claim 8, wherein the topology converter searches for a connector of a new conflict links tuple in the list of existing tuples.

15. (currently amended) A computer-readable medium having computer-executable instructions for performing a method for mapping a network topology by

identifying changes between an existing topology and a new topology in a network having a interconnected nodes, the method comprising:

creating a plurality of tuples for a topology of a network, wherein the tuples represent nodal connections of the network and wherein each of the tuples comprises a host identifier, interface information, and a port specification;

~~converting an existing topology into a list of existing tuples that represent existing nodal connections;~~

receiving new tuples that represent new nodal connections;

comparing the list of existing tuples with the new tuples to identify changes to the topology; and

updating a topology database with a new topology based on the comparing.

16. (currently amended) The medium method of claim 15, wherein a topology converter receives the new tuples from a connection calculator that calculates connections between nodes.

17. (currently amended) The medium method of claim 15, wherein the step of comparing comprises identifying duplicate tuples that appear both in the list of existing tuples and in the new tuples, and maintaining a current status of the topology for these tuples.

18. (currently amended) The medium method of claim 15, wherein the step of comparing comprises identifying a swapped port condition on a connector.

19. (currently amended) The medium method of claim 15, wherein the step of comparing comprises searching for a host of a new singly-heard host link tuple or a new multi-heard host link tuple in the list of existing tuples.

20. (currently amended) The medium method of claim 15, wherein the step of comparing comprises searching for a connector of a new conflict links tuple in the list of existing tuples.



**REMARKS**

Claims 1-3 and 5-20 are pending. Claims 1-20 are rejected. Claims 1-7, 9, and 15-20 are rejected under 35 U.S.C. § 112, ¶ 1 as lacking enablement. Claims 8, 10, and 12-14 are rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 4,644,532 to George, et al. (hereinafter “George”). Claim 11 is rejected under 35 U.S.C. § 103(a) as being unpatentable over George in view of U.S. Patent No. 5,023,873 to Stevenson, et al. (hereinafter “Stevenson”). By this amendment, claims 1, 8, 9, and 15-20 are amended. Claim 4 is canceled. No new claims are added. No new matter is added. Reconsideration and allowance is requested.

Claim 1 is amended to recite the step of “creating a plurality of tuples for a topology of a network” in place of the current step of “converting....” Claim 1 is rejected under section 112 for lacking enablement for the term “converting.” Applicant respectfully disagrees with the examiner’s assertion that the term “converting” lacks enablement. The converting process is described with respect to Figures 18a-b, also referred to as the “morph topo” phase of the process. However, the above amendments replace the term “converting” in favor of the step of “creating” to more particularly claim the invention. Support for this amendment is found, for example, on pages 16-17 of the specification and in Figures 17 and 18a-b. As amended, claim 1 overcomes the section 112 rejection. Reconsideration is requested.

Claim 1 is further amended to define the tuples as comprising “a host identifier, interface information, and a port specification.” As amended, claim 1 overcomes the cited references. The cited references do not teach or suggest a tuple containing these items. As amended, claim 1 is allowable. Claims 2-3, and 5-7 depend from claim 1 and for this reason and the other limitations they recite, are also allowable.

Claim 4 is hereby canceled without prejudice or disclaimer.

Claim 8 has been amended to recite that the tuples comprise “a host identifier, interface information, and a port specification” and that the system compares new tuples with the existing topology by “comparing the host identifiers, the interface information, and the port specifications.” The cited references do not teach or suggest these particular elements of the tuple. Nor do the cited references teach or suggest comparing these elements to identify changes to the network topology. In contrast, George monitors data in the topology database

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

## LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

## FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

## E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.