US 20070226375A1

(19) United States
(12) Patent Application Publication
Chu et al.
(10) Pub. No.: US 2007/0226375 A1
(43) Pub. Date: Sep. 27, 2007

(54) PLUG-IN ARCHITECTURE FOR A NETWORK STACK IN AN OPERATING SYSTEM

(76) Inventors: **Hsiao-Keng J. Chu**, Palo Alto, CA (US); **Darrin P. Johnson**, Mountain View, CA (US); **Ka-Cheong Poon**, Kowloon (HK)

Correspondence Address:
SUN MICROSYSTEMS INC.
C/O PARK, VAUGHAN & FLEMING LLP
2820 FIFTH STREET
DAVIS, CA 95618-7759 (US)

(21) Appl. No.: **11/388,438**

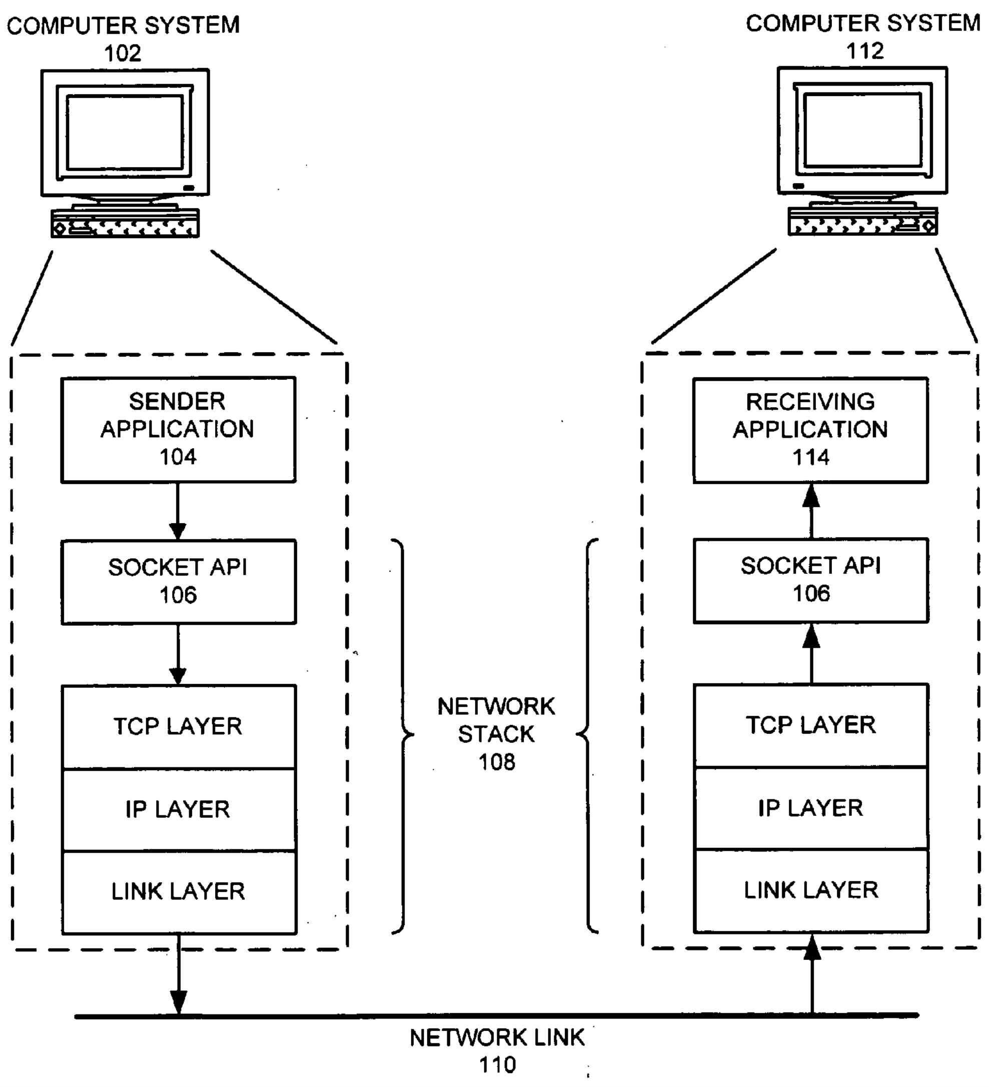
(22) Filed: **Mar. 23, 2006**

Publication Classification

(51) **Int. Cl.**
G06F 15/16 (2006.01)
(52) **U.S. Cl.** **709/250**; 709/228; 709/227

(57) **ABSTRACT**

One embodiment of the present invention provides a plug-in architecture for a network stack in an operating system. The network stack includes a set of functions configured to modify a set of parameters that are likely to change based on the network environment. The architecture includes a plug-in framework within the network stack that allows the set of functions to be dynamically changed in order to change the TCP behavior of the network stack to suit the network environment.



AKAMAI

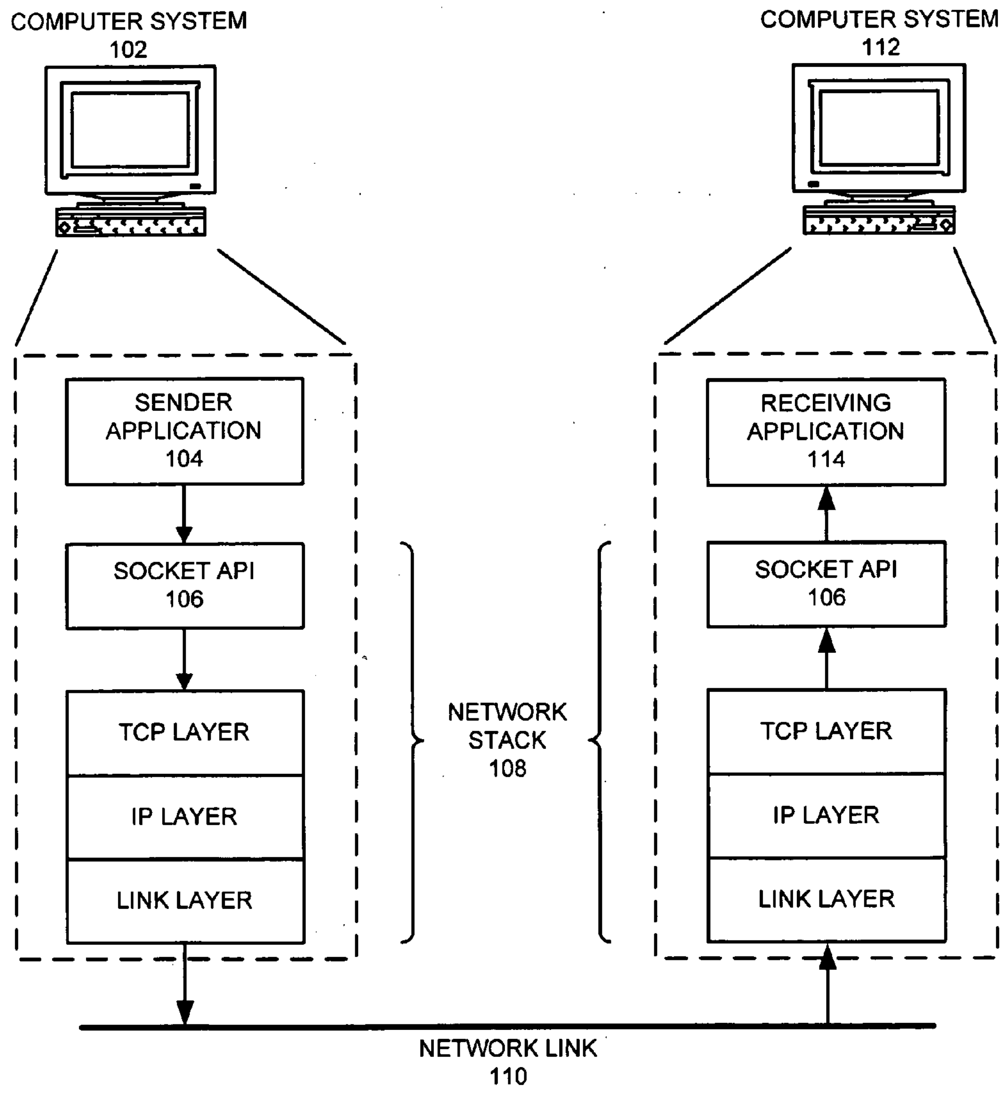

COMPUTER SYSTEM
102
COMPUTER SYSTEM
112
SENDER APPLICATION
104
SOCKET API
106
TCP LAYER
IP LAYER
LINK LAYER
RECEIVING APPLICATION
114
SOCKET API
106
TCP LAYER
IP LAYER
LINK LAYER
NETWORK STACK
108
NETWORK LINK
110

FIG. 1

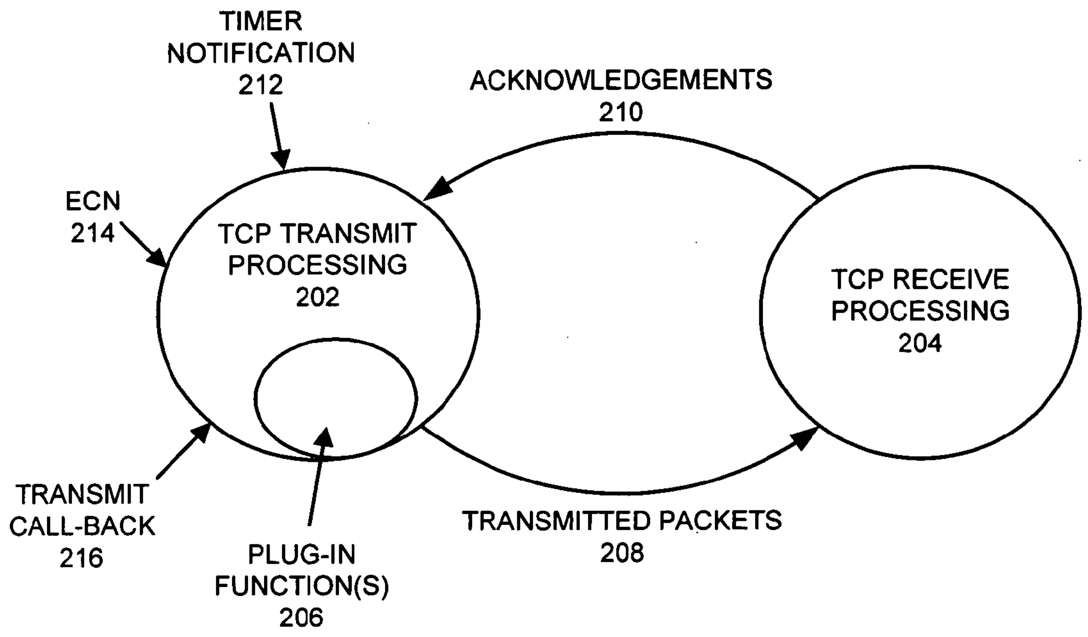

TIMER NOTIFICATION 212
ACKNOWLEDGEMENTS 210
ECN 214
TCP TRANSMIT PROCESSING 202
TCP RECEIVE PROCESSING 204
TRANSMIT CALL-BACK 216
TRANSMITTED PACKETS 208
PLUG-IN FUNCTION(S) 206

FIG. 2

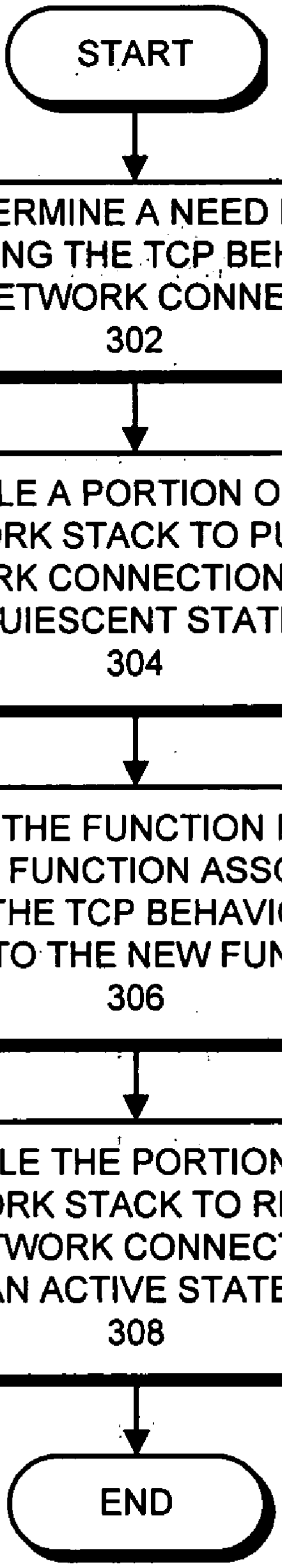

START
DETERMINE A NEED FOR CHANGING THE TCP BEHAVIOR OF A NETWORK CONNECTION
302
DISABLE A PORTION OF THE NETWORK STACK TO PUT THE NETWORK CONNECTION INTO A QUIESCENT STATE
304
CHANGE THE FUNCTION POINTER FOR THE FUNCTION ASSOCIATED WITH THE TCP BEHAVIOR TO POINT TO THE NEW FUNCTION
306
RE-ENABLE THE PORTION OF THE NETWORK STACK TO RETURN THE NETWORK CONNECTION TO AN ACTIVE STATE
308
END

FIG. 3

PLUG-IN ARCHITECTURE FOR A NETWORK STACK IN AN OPERATING SYSTEM

RELATED APPLICATION

[0001] The subject matter of this application is related to the subject matter in a co-pending non-provisional application by the same inventors as the instant application and filed on the same day as the instant application entitled, "Method and Apparatus for Dynamically Changing the TCP Behavior of a Network Connection," having serial number TO BE ASSIGNED, and filing date TO BE ASSIGNED (Attorney Docket No. SUN06-0663).

BACKGROUND

[0002] 1. Field of the Invention

[0003] The present invention generally relates to computer networks. More specifically, the present invention relates to a plug-in architecture for a network stack in an operating system.

[0004] 2. Related Art

[0005] The transmission control protocol (TCP) is part of the core Internet protocol which is used to transfer data between computing devices. The goal of TCP is to transfer data from an application on a computing device through a shared network resource to a second device as quickly, efficiently, and reliably as possible, despite potential contention and congestion.

[0006] While the basic operation of TCP has not changed dramatically since the initial publication of the standard in 1981, the protocol has been forced to evolve in response to changing network conditions such as new link types (e.g., wireless networks) and higher bandwidth wired networks. Substantial ongoing research on congestion control and avoidance has resulted in numerous TCP congestion control techniques, such as Reno, New Reno, Vegas, HS-TCP, Fast TCP, S-TCP, and Bic-TCP. However, such congestion control techniques add substantial complexity to TCP and the network stack. Furthermore, end-to-end links can traverse numerous networks with diverse characteristics, and no single congestion control approach encompasses the wide range of modern networks.

[0007] Hence, what is needed are architectures and methods that facilitate congestion control for TCP without the limitations of existing approaches.

SUMMARY

[0008] One embodiment of the present invention provides a plug-in architecture for a network stack in an operating system. The network stack includes a set of functions configured to modify a set of parameters that are likely to change based on the network environment. The architecture includes a plug-in framework within the network stack that allows the set of functions to be dynamically changed in order to change the TCP behavior of the network stack to suit the network environment.

[0009] In a variation on this embodiment, the parameters include:

[0010] a round-trip time ("RTT"), which is the time it

[0011] a congestion window ("cwnd"), which specifies the number of data packets that can be transmitted without having received corresponding acknowledgement packets; and/or

[0012] a slow-start threshold ("ssthresh"), which determines how the size of the congestion window increases.

[0013] In a variation on this embodiment, changing the set of functions changes the transmit and receive characteristics of the network stack, thereby changing the congestion-control technique for the network stack.

[0014] In a further variation, the set of functions are triggered by events that include:

[0015] the receipt of a positive acknowledgement indicating that a packet was received;

[0016] the receipt of negative acknowledgements indicating that packets may have been lost;

[0017] the receipt of a selective acknowledgement that identifies received packets;

[0018] the expiration of a timer;

[0019] the elapse of a round-trip time interval;

[0020] a call-back occurring either before or after a packet transmission; and

[0021] the receipt of an explicit congestion notification (ECN).

[0022] In a further variation, triggering an event prompts the set of functions to update the set of parameters.

[0023] In a variation on this embodiment, the network stack maintains a set of generic state information, and the set of functions maintains a set of state separate from the set of generic state information. The set of functions can access the set of generic state information.

[0024] In a variation on this embodiment, the set of functions is implemented as a dynamically loadable kernel module.

[0025] In a variation on this embodiment, changing the set of functions allows the network stack to dynamically change TCP behavior and thereby transmit efficiently across diverse and changing network environments.

BRIEF DESCRIPTION OF THE FIGURES

[0026] FIG. **1** illustrates two computer systems communicating over a network link in accordance with an embodiment of the present invention.

[0027] FIG. **2** illustrates TCP transmit and receive interactions in accordance with an embodiment of the present invention.

[0028] FIG. **3** presents a flow chart illustrating the process of changing the TCP behavior of a network connection in accordance with an embodiment of the present invention.

DETAILED DESCRIPTION

[0029] The following description is presented to enable

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.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE

fastcase® Smarter legal research.