PCT
WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
H03M 7/30, H04L 23/00, G06F 7/00, 7/06, 7/22

A1

(11) International Publication Number: WO 96/25801

(43) International Publication Date: 22 August 1996 (22.08.96)

(21) International Application Number: PCT/AU96/00081

(22) International Filing Date: 15 February 1996 (15.02.96)

(30) Priority Data:

PN 1232	17 February 1995 (17.02.95)	AU
PN 2392	12 April 1995 (12.04.95)	AU

(71) Applicant *(for all designated States except US)*: TRUSTUS PTY. LTD. [AU/AU]; 200 East Terrace, Adelaide, S.A. 5000 (AU).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: WILLIAMS, Ross, Neil [AU/AU]; 16 Lerwick Avenue, Hazelwood Park, S.A. 5066 (AU).

(74) Agent: MADDERNS; 1st floor, 64 Hindmarsh Square, Adelaide, S.A. 5000 (AU).

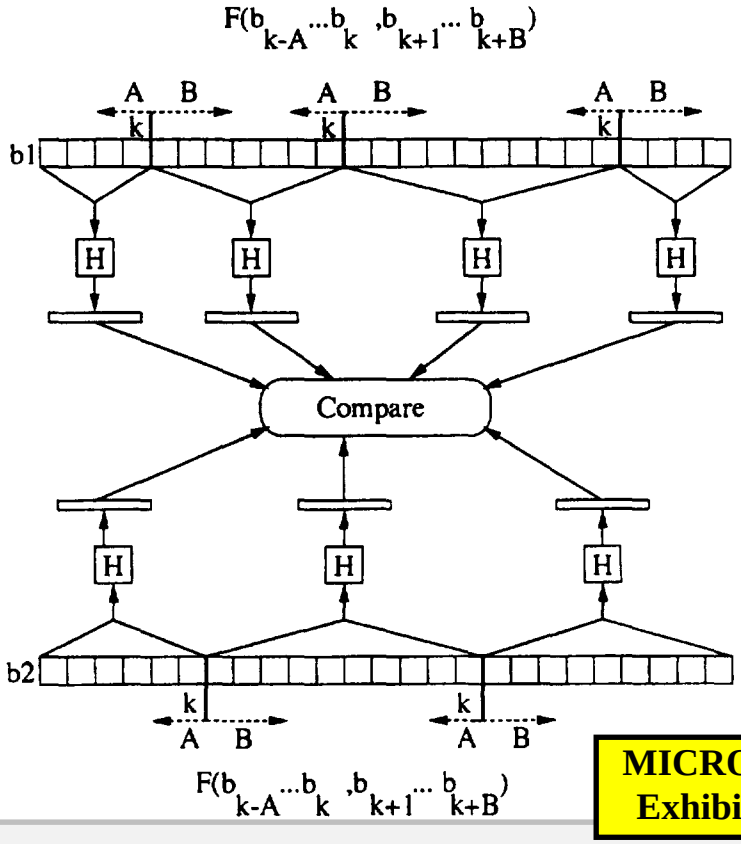
(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, US, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AZ, BY, KG, KZ, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.
With amended claims and statement.

(54) Title: METHOD FOR PARTITIONING A BLOCK OF DATA INTO SUBBLOCKS AND FOR STORING AND COMMUNICATING SUCH SUBBLOCKS

(57) Abstract

This invention provides a method and apparatus for detecting common spans within one or more data blocks by partitioning the blocks (figure 4) into subblocks and searching the group of subblocks (figure 12) (or their corresponding hashes (figure 13)) for duplicates. Blocks can be partitioned into subblocks using a variety of methods, including methods that place subblock boundaries at fixed positions (figure 3), methods that place subblock boundaries at data-dependent positions (figure 3), and methods that yield multiple overlapping subblocks (figure 6). By comparing the hashes of subblocks, common spans of one or more blocks can be identified without ever having to compare the blocks or subblocks themselves (figure 13). This leads to several applications including an incremental backup system that backs up changes rather than changed files (figure 25), a utility that determines the similarities and differences between two files (figure 13), a file system that stores each unique subblock at most once (figure 26), and a communications system that eliminates the need to transmit subblocks already possessed by the receiver (figure 19).



MICROSOFT
Exhibit 1002

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	**GB**	United Kingdom	**MW**	Malawi
AT	Austria	**GE**	Georgia	**MX**	Mexico
AU	Australia	**GN**	Guinea	**NE**	Niger
BB	Barbados	**GR**	Greece	**NL**	Netherlands
BE	Belgium	**HU**	Hungary	**NO**	Norway
BF	Burkina Faso	**IE**	Ireland	**NZ**	New Zealand
BG	Bulgaria	**IT**	Italy	**PL**	Poland
BJ	Benin	**JP**	Japan	**PT**	Portugal
BR	Brazil	**KE**	Kenya	**RO**	Romania
BY	Belarus	**KG**	Kyrgystan	**RU**	Russian Federation
CA	Canada	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CF	Central African Republic			**SE**	Sweden
CG	Congo	**KR**	Republic of Korea	**SG**	Singapore
CH	Switzerland	**KZ**	Kazakhstan	**SI**	Slovenia
CI	Côte d'Ivoire	**LI**	Liechtenstein	**SK**	Slovakia
CM	Cameroon	**LK**	Sri Lanka	**SN**	Senegal
CN	China	**LR**	Liberia	**SZ**	Swaziland
CS	Czechoslovakia	**LT**	Lithuania	**TD**	Chad
CZ	Czech Republic	**LU**	Luxembourg	**TG**	Togo
DE	Germany	**LV**	Latvia	**TJ**	Tajikistan
DK	Denmark	**MC**	Monaco	**TT**	Trinidad and Tobago
EE	Estonia	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**UG**	Uganda
FI	Finland	**ML**	Mali	**US**	United States of America
FR	France	**MN**	Mongolia	**UZ**	Uzbekistan
GA	Gabon	**MR**	Mauritania	**VN**	Viet Nam

Method For Partitioning A Block of Data Into Subblocks And For Storing And Communicating Such Subblocks

INTRODUCTION

The present invention provides a method and apparatus for identifying identical subblocks of data within one or more blocks of data and of communicating and storing such subblocks in an efficient manner.

BACKGROUND

Much the massive amount of information stored, communicated, and manipulated by modern computer systems is duplicated within the same or a related computer system. It is commonplace, for example, for computers to store many slightly differing versions of the same document. It is also commonplace for data transmitted during a backup operation to be almost identical to the data transmitted during the previous backup operation. Computer networks also must repeatedly carry the same or similar data in accordance the requirements of their users.

Despite the obvious benefits that would flow from a reduction in the redundancy of communicated and stored data, few computer systems perform any such optimization. Some instances can be found at the application level, one example being the class of incremental backup utilities that save only those files that have changed since the most recent backup. However, even these utilities do not attempt to exploit the significant similarities between old and new versions of files, and between

files sharing other close semantic ties. This kind of redundancy can be approached only by analysing the contents of the files.

The present invention addresses the potential for reducing redundancy by providing an efficient method for identifying identical portions of data within a group of blocks of data, and for using this identification to increase the efficiency of systems that store and communicate data.

SUMMARY OF THE INVENTION

To identify identical portions of data within a group of blocks of data, the blocks must be analysed. In a simple aspect of the invention, the blocks are divided into fixed-length (e.g. 512-byte) subblocks and these subblocks are compared with each other so as to identify all identical subblocks. This knowledge can then be used to manage the blocks in more efficient ways.

Unfortunately, the partitioning of blocks into fixed-length subblocks does not always provide a suitable framework for the recognition of duplicated portions of data, as identical portions of data can occur in different sizes and places within a group of blocks of data. Figure 1 shows how division into fixed-size subblocks fails to generate identical subblocks in two blocks whose only difference is the insertion of a single byte ('X'). A comparison of the two groups of subblocks would reveal no identical pairs of subblocks.

In a more sophisticated aspect of the invention, the blocks are partitioned at boundaries determined by the content of the data itself. For example, the block could be divided at each point at which the preceding three bytes hash to a particular constant value. Figure 2 shows how such a partitioning could turn out, and contrasts it with a fixed-length partitioning.

The fact that a partitioning is data dependent does not imply that it must incorporate any knowledge of the syntax or semantics of the data. So long as the boundaries are positioned in a manner dependent on the local data content, identical subblocks are likely to be formed from identical portions of data, even if the two portions are not identically aligned relative to the start of their enclosing blocks (Figure 3).

Once the group of blocks has been partitioned into subblocks, the resulting group of subblocks can be manipulated in a manner that exploits the occurrence of duplicate subblocks. This leads to a variety of applications, some of which are listed below. However, the application of a further aspect of the invention leads to even greater benefits.

In a further aspect of the invention, the hash of one or more subblocks is calculated. The hash function can be an ordinary hash function or one providing cryptographic strength. The hash function maps each subblock into a small tractable value (e.g. 128 bits) that provides an identity of the subblock. These hashes can usually be manipulated more efficiently than their corresponding subblocks.

Some applications of aspects of this invention are:

Fine-grained incremental backups: Conventional incremental backup technology uses the file as the unit of backup. However, in practice many large files change only slightly, resulting in a wasteful re-transmission of changed files. By storing the hashes of subblocks of the previous versions of files, the transmission of unchanged subblocks can be eliminated.

Communications: By providing a framework for communicating the hashes of subblocks, the invention can eliminate the transmission of subblocks already possessed by the receiver.

DOCKET ALARM

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.