533333;: ‘ . PATENT
NUMBER , _- f: NUMBER

SERIALNUMBER'. . " '.. " ""-.GROUP'ART'UNIT'
- 33/553.3?3‘ 11/33/33. -;3. ~" 3 '~ ”' ' 3 3307

wammMILL..HUu3T3N. TX; 33013 3. mammHiLu. 33WMMHND, TX;
R. MDRE 33.. HDU3TQN.‘Txg 33v H. 333313. Hmu3333. 3x.

:xé;:§;:{~:'

3:33:33 m;3--I...3\3'.1:3 ' ~ 3.j fl, LIV/IIIi3.‘/"’EE3III{3 3133333

3333-33"'L.. 3333 {'3 '3 33333393

. i "

‘53:!333:E5I|2'=33\3 3:13... 133333553 3;.,. 1332333333533 1:2339‘33'33‘33'3? ‘ "§oretgn priomy claimed (3 yes 'En6
. 8'5 USC119 commons me! C] yes no‘ _ "V.°f333°d WW3'ACkf'IOMWQ'ed Examine'r‘snmas. 3 " ".. 3 3 3.. ".' ' " " -331333152:Fan‘s":

31333;: L. 33\7‘33333 3' ‘ ‘ - ‘ ., . 3 ~ ‘

. xFILINGu, FEE ATTORNEYS,
"cwms necawrso DOCKET NO,

3:4 3.- I.) I... 333: H (-3 331 T

.

,,.

' §Y$TEMjs METHHD me KINTHIUUTLM‘HTHRQHL MANAGEMPNF HN N333u3333

L.:,..m3w.:mz3::YRIHYH:35n9 3nqu 0393333 12463331935. lé/flflfé . ‘,, HU.S. DEPT. OF UOMU./PA‘F &TM"~“PTOv436L (Bev12'29..-..rw...............W.... “......“m........-.....,..,,..

7 PARTS OF APPLICATION . ~. I W/.)51‘7 c
”LED SEPARATELY _ ‘ ' _ ‘ _. . Anlications Examiner _
‘ NOTICE OF ALLOWANCE MAILED ' ' , ' ' CLAIMS ALLOWED
' Total Claims,‘w 'P'rint Ciaim 4

Ass3stant Examiner » I

4-. 3 . 3333.33RYE"’\~ ‘. 3 ~. EXHIBIT
QF‘Soncs‘A LAW/[61>

Z 00 BL

313,33"?3(‘0

an

. Q.0.0
£9

UTILITY ‘ V "" PATENT DATE PATENT
SERIAL NUMBER NUMBER

SERtAL NUMBER ‘ FILING DATE cuAss " ' SUBCLASS ‘ _ ' GROUPARY UNIT

08/085,596 07/01/93 ' ."‘>»/\\v> ”1. 2307

JAMES R. WDUDHILL, HOUSTON, TX; LOUIS R. WOODHILL, RICHMOND, TX; AppLitANTs WILLIAM R. MORE JR., HOUSTON, TX: JAY H. BERLIN HOUSTON, TX.‘
*$CQNTINUING Dng****t*#****$***$*****\
VERIFIED

4112mm W8 tr

“*FOREIGN/PCT APPLICATIONS$$$$$$******
VERIFIED

mm- mi

FOREIGN FILING LICENSE GRANTED 10/05/93

Foralgn priorlty claimed , ATTORNEV'S
35 USC 119 conditions mat D vs: R no DOCKET NO.

Verttled and Acknowledged Examlheri::nitials

CHR I STDPHER H. GEBHQRDT
K I RKP‘ATR I CK 8c LOCKHART

1500 OLIVER BUILDING
P I TTSBURGH,, PA 1 5222

SYSTEM AND METHOD FOR DISTRIBUTED STORAGE M NAGE EN ON NETwORKED

COMPUTER SYSTEMSAW (23WW

US. 3591"

M
07 COMMaVI‘. 6: TM otflco~ Fro-43m. (70V. 10-78)

PARTS OF APPLICATION . _ ”LED SEPARATELY; , Antications Examiner
NOTICE OF ALLOWANCE MAILED ‘ '., ‘ CLAIMS ALLOWED
» ‘ Total Claims Print Claim

Assistant Examiner '

iSSUE FEE . ' DRAWlNG i
; Amount Due Date Paid Sheets Drwg. . Print Fig.

ISSUE I
BATCH

Primary Examiner NUMBER
PREPARED FOR ISSUE, V

WARNING: The information diSclosed herein may be restricted. Unauthorized disclosure may be prohibited

by the United States Code Title 35, Sections 122, 181 and,368. Possession outside the U.S. n-p.w.a o T.-.A..m.ut. Am-.. in waa‘u‘uOn—J 6—. «Hohhdwmd Amn‘hvlnnk ”ant finn0vn~3¢~nn “at”

08/085,596

SYSTEM AND METHOD FOR DISTRIBUTED STORAGE MANAGEMENT ON NETWORKED COMPUTER SYSTEMS

Transaction History

Date Transaction Descri ition
O7-0l-l993 initial Exam Team nn

08—02nl993 Notice Mailed-—A lication Incomlete--Filin; Date Assined
10—01—1993 A lication Is Now Comm
10-184993 A, lication Catured on Microfilm
1’0-29~l993 Case Docketed to Examiner in GAU
Case Docketed to Examiner in GAU
10-20— 1994 Non~Fina1 Re'ection
10914994 Mail NonvFinal Re' ection

02-23-1995 Information Disclosure Statement (IDS) Filed
02-23-1995 Information Disclosgifitatement IDS Filed
02-23-1995 Res onse after Non—Final Action
02'23—1995 Re uest for: Extension of Time - Granted
03-15wl995 Date Forwarded to Examiner

05—08-1995 Final Rejection
051 14995 Mail Finalwlggjection (FIOL — 326)
ll-O9~l995 Reuest for Extension of Time ~ Granted
01-11-1996 Aband. for Failure to Res 0nd to O A
01—16-1996 Mail Abandonment for Failure to Resond to Office Action

02~Ol-l996 Abandonment for Purposes of Filing an FWC - File Combined with Child
A., lication

5,649,196

SYSTEM & METHOD FOR DISTRIBUTED STORAGE MANAGEMENT ON NETWORKED COMPUTER SYSTEMS USING BINARY OBJECT

01~23—l996

01-02-1997

Date Transaction Descrigtion I
12—14-1995

10—24-1996 Prelimina Amendment
12—04-1996 Case Docketed to Examiner in GAU

IDENTIFIERS

Transaction History

Initial Exam Team nn
A lication Ca tured on Microfilm

Mail Notice of Allowance
01w02-l997
' 02-05.1997
02—28-1997

Amendment after Notice of Allowance Rule 312

‘ 03-06-1997

Notice of Allowance Data Verification Comleted

Date Forwarded to Examiner

Mail Res onse to 312 Amendment PTO—271}

06—09—1 997
07-15—1997
04—02-2002
0403-2002

03-06~l 997 Res onse to Amendment under Rule 312

03—18-1997 Mailroom Date of Drawings)
i03—20—l997 Issue Fee Pa ment Verified

O4~l4-l997 Drawing 5 Received at Publications
04*15-1997 Drawin s Matched to A.lication

04~18~1997 Drawingts) Processing Comgleted

01-25-2005 Applicant Has Filed a Verified Statement of Small Entity Status in Compliance

Issue Notification Mailed
Recordation of Patent Grant Mailed
Corresondcnce Address Chane

Change in Power of AttorngflMaX Include Associate POA)

with 37 CFR 1.27

Date ? uaxe \ 1
R i d "

5333,21: CONTE , F REGWH) '
. OCT 2 1. W95
. ' ~ GROUPZSOO'
1. Application ‘. papers.

—- l..

312. 4 ' ‘ '

W14.W m

Wm@ M

- 1 am WW

1 ' 20;. ‘ .. ‘ . '

Mam

.

4

,

"

”

, 21.
22.

25.

2 4'W

26.

‘ ‘ 4

-

, .
'

_._._....._‘._

.

..

W
W
W

W
W

. . . . _._...W

‘

‘

i

’ '90

27.
2 8.W

.

h

‘

L

. .

W
W

.

Date
Entered
or

Counted

u

CONTENTS '

Date '
Received

Magfed
RECEVED.

JAN 2 5 39%

GROUP 3200 N

mm

PPLICATION SERIXC9bWaéfi 2' ‘ 94..

03 55’5“.374 I;
gileCA ‘ “SKAME (PLEASE PRINT) ' ‘ 3/ V

ASSlSTANT EXAMINER (PLEASE STAM ORPRINT FULL NAME) " "T ' //‘7/'

. ' ', 3 us EPARTMENT'OFCMMEHCE'
' (REM-91) I ISSUE CLASSIFICATION SLIP, - Ememmommwinmncs \

W—

m...— ~ /. W

m -

—_-—

” mum

- —

spec-HAND II. “
a , ..

”-

INDEX OF CLAIMS.

‘ Date , Ciaim Date
2 a E

3‘I WIIIII l g III-Ill
i-fi'aEEEEEE .=====---

..:-====.-= ..===:==== 54
IIIIIIIIII 55 III—III

$9 III-II 59 IIII
10 IIIII EII
an -n .61 I I

"EA‘WII 14 I ~ I .6463 E-I-I

‘ -IMMII

WEI ””4". II:—
III-

a ’ m...» 68 I s as n 59 - i

III

_0. all 7° - 4
71 -
22 II 72

IIIIIII , 73 -II
I I I III-

26 I III—I _ 3; -=-—-=— -;7”-I=II

IIIIIIIII- W . II—IIIII '

-..=.-.====. =m~===-- 32 - --- +

i’awmgs"6333:?“ assumed I 825" I

.

s ,IIII.a ‘2‘: swasssssssss II—II
I 34 III-II A ,onea! III-II

, ::.=====.. =%.-====.--
III-III . III-III

III-III IKE-III

ZS BEEP-Pu: "BEE-.-
43 IIIIIIII IIIIIIII
~44 IIIIIIII IIIIIII

INDEX OF CLAIMS

Claim Date

.6 jg 2,2 ~
E g 30
:.. u I
ail-II

E====--

III-III
IlIIIIIII ~

IIII-I I
III-III
' I II

I —

I II

:

'2728 ‘1= III=II I II II
: I

u 31:= =-.I= 2’ 2291111.;é1212iiiiiéliliiigll3313;: —.--.

,. m . (hroug num era ancee

{w:1::'"""“""'“':::::::::::: $332333; _. Me uvence

Q, fl-= é:t:::1:.,....‘.‘,.‘,‘.A.:::::::::83:3;
-' IIII

; III-II

III-IIII
III-I i

III-III- ' 43

POSITION

mm—

TYPIST , —

- -

VERIFIER - Ea

' -

-

. I

SPEC. HAND

FiLE MAINT.
DRAFTING.

J...

160 KM 07/22/93 08085596

[JR/085598

PATENT APPLICATION SERIAL NO.

U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE
‘ FEE RECORD SHEET

1 101 932.00 OK 93085I

fly/0&&$:96
m

PA TENT

Docket No "/ 93085 W

Commissioner of Patents and Trademarks CW,
Washington, 0.0. 20231

NEW APPLICATION TRANSMITTAL

Transmitted herewith for tiling is the patent application of

lnventor(s): Z‘)0 fi/fiufifl/
James~WR. Woodhill et a1..W'

WARNING Patent must be applied lor in the namels) oi all 0! the actual inventor(s) 37 OFF? 1 41(a) and 1.553(1))

For (title):

SYSTEM AND METHOD FOR DISTRIBUTED STORAGE MANAGEMENT ON NETWORKED COMPUTER SYSTEMS

1‘ Type of Application

This new application is for a(n) (check one applicable item below): ‘
:53] Original ’
Ci Design
[3 Plant
WARNING: Do not use this transmittal for a completion in the US. of an international Application under 35 U. S. C. 371(c)(4) unless the international Application is being filed as a divisional. continualionor continuation-impart application.
NOTE: It one of the following 3 items apply then complete and attach ADDED PAGES FOR NEW APPLICA-TION TRANSMITTAL WHERE BENEFIT OF A PRIOR U. 8. APPLICATION CLAIMED.

a Divisional

[j Continuation
[j Continuation—impart (CIP)

-w

CERTIFICATION UNDER 37 CFR 1.10

l hereby certify that this New Application Transmittal and the doojumTts1reterredt<§61953enclosed therein are being deposited with the United States Postal Service on this date in an envelope
as “Express Mail Poet. Ollice to Addressee“ Mailing Label Number H37 1 97 4 7 addressed, to the: Commissioner of Patents and Trademarks. Washington, 0.0. 20231.
Beth H. Retort

NOTE: Each paper or lee referred to as enclos- ere/n has the number of the "Express Mail" mailing label placed thereon prior to mailing. 37 CFR 1.700)).

(Application Transmittal [4-1]-—~page 1 of 7)

2. Benefit of Prior us. Application(s) (35 use 120)
NOTE: It the new application being transmitted Is a divisional, continuation or a continuation-impart ol'a parent case, or where the parent base is an International Application which designated the U.S.. then check
the following item and complete and attach ADDED PAGES FOR NEW APPLlCA TION TRANSMiTTAL
WHERE BENEFIT OF PRIOR U.S. APPLICA TION(S) CLAIMED.

[j The new application being transmitted claims the benefit of prior U.S. applica-tionts) and enclosed are ADDED PAGES FOR NEW APPLICATION TRANSMIT-
TAL WHERE BENEFIT OF PRIOR U.S. APPLICATION(S) CLAIMED.

3. Papers Enclosed Which Are Required For Filing Date Under 37 CFR 1.53(b) (Reg-ular) or 37 CFR 1.153 (Design) Application

”3.3.." Pages of specification

. 8”, Pages of claims

ml.” Pages of Abstract-

“.13 Sheets ct drawing

[3 formal

g] informal
WARNING: DO NOT submit original drawings. A high quality copy of the, drawings should be supplied when filing a patent application. The drawings that are submitted to the Ollice must be on strong, white,
smooth, and non~shiny paper and meet the standards according to § 1.84. It corrections to the
drawings are necessary, they should be made to the original drawing and a high-quality copy at
the corrected original drawing then submitted to the Office. Only oneicopy ls required or de-
sired. Comments on proposed new 37 cm 1.84. Notice of March 9, 1988 (1990 0.6. 5762).
NOTE: "Identifying indicia such as the serial number, group and unit, title of theinvention, attorney’s docket number, inventors name, number of sheets, etc, not to exceed 23/4 inches (7.0 cm.) in width may be placed in a centered location between the side edges within three fourths inch (19.1 mm.) of the top
edge. Either this marking technique on the front of the drawing or the placement, although not pre-
ferred, of this information and the title of the invention an the back of the drawings is acceptable. " Pro-
posed 37 CFR 1.234(1). Notice of March 9, 1988 (1090 0.6. 57-62).

4. Additional papers enclosed
Preiiminary Amendment
Information Disclosure Statement.
Form PTO~1449

Citations

Declaration of Biological Deposit

Authorization of AttorneyIs) to Accept and Follow Instructions from Representa-
tive

Special Comments
Other

CID DECIDED

(Application Transmittal [4-11—page 2 of 7)

5. Declaration or oath

C] Enclosed

executed by (check all applicable boxes)
[3 lnventor(s).

iii legal representative of inventorts). 37 CFR 1.42
or 1.43

D joint inventor or person showing a proprietary
interest on behalf of inventor who refused to
sign or cannot be reached.

D this is the petition required by 37 CFR 1.47 and the statement
required by 37 CFR 1.47 isalso attached. See item 73 lye/aw tor
lee,
Not Enclosed.

WARNtNG:,Where the tiling is a completion in the (1.3 of an International Application but where a declaration is not available or where the completion 0/ the US. application contains sub/eat matter in addition to the International Application the application may be treated as a continuation or continuation-impart, as the
case may be. utilizing ADDED PAGE FOR NEW APPLICA TION TRANSMITTAL WHERE BENEFIT OF PRIOR
US. APPLICATION CLAIMED

Ci Application is made by a person authorized under 37 CFR t4l (c) on behalf

Of all the above named inventorls). {The declaration or oath, along wrlh the Surcharge root/tied by 37 OFF? 1 H979) can be tiled subsequent/y)
NOTE It IS important that all the correct invenIOits) are named tor tiling under 37 CFR 1.4 1(0) and t.53{bt.

[It Showing that the filing is authorized (Not required unless calledinlo question 37 CFR l.4 ltd).

6. lnventorship Statement
WARNING: It the named Inventors are each not the inventors 0/ all the claims an explanation. including the ownership oi the various claims at the time the last claimed invention was made. should be submitted

The inventorship for all the claims in thisapplication are:
lit The same

or i_ ,.

[I] Are not the same. An explanation, including theownership of the various claims at the
time the last claimed invention vies made,
Cl is submitted.
D will be submitted.

7. Language
NOTE: An application including a signed oath or declaration may be filed in a language other than English.
A verified English translation of the non-English Ianguage'application and the processing tea at $130.00 required by 37 CFR t. t7{t<) is required to be filed with the application or within such time
as may be set by the Office. 37 CPR 1.52M).
NOTE: A non-English oath or declaration in the form provided or approved by the PTO need not be translated.
37 CFR 7.69m.

English
l3 ‘ non—English

C] the attached translation is a verified translation. 37 CFR 1.52ld).
(Application Transmittal [ct-llwpage 3 of 7)

8. Assignment
EX An assignment of the invention to LEGENT C03? oration

D is attached. A separate U "COVER SHEET FOR ASSlGNMENT (DOCU-
MENT) ACCOMPANYING NEW PATENT APPLICATION" or C] FORM PTO
‘ 1906 is also attached.
X] will follow.
NOTE: "ii an assignment is submitted with a new application, send two separate letters-one for the
application and one for the assignment." Notice of May 4, 1990 (7114 0.6. 77-78).
9. Certified Copy
Certified copyties) ot applicationls)

(country). (appint no.) 3 (filed)

(country) (appln. no.) (med).

(country) ‘ (appln. no.) . (filed)
from which priority is claimed
E3 ls(are) attached.
[I] will follow.

'NOTE: The foreign application forming the basis for the claim for priority must he telerred to in the oath or
declaration, 37 CH? 165(3) and 7.63. '
NOTE: This item is for any foreign priority for which the application being filed directly relates. if any parent
(1.8. application or InternationaiAppiication from which this application claims benefit under 35 Ms. C.
120 is itself entitled to priority lrom a prior foreign application then complete item 18 on the ADDED
PAGES FOR NEW APPLiCA TION TRANSMITTAL WHERE BENEFIT OF PRIOR us. APPLICA-

TiONtS) CLAIMED.
)0. Fee Calculation (37 CFR 1.16)
At [34 Regular application

CLAIMS AS FILED

Number filed Number Extra Rate Basic Fee
37 CFR 116(8)
$710.00
Total .

Claims (37 CFR 1.16(c)) 20 ~20: 0 x $ 22.00 - 0

independent . Claims (37 CFR 1.1603» 6 was 3 x $ 74.00 222'00
Multiple dependent claim(s_), if any
(37 CFR' 1.16(d)) ‘ ‘ $230.00

E3 Amendment cancelling extra claims enclosed.
Cl Amendment deleting multiple—dependencies enclosed.
[3 Fee for extra claims is not being paid at this time.
NOTE: If the fees for extra claims are not paid on filing they must be paid or the claims cancelled by
amendment, prior to the expirationiof the time‘per/od set for response by the Patent and Trademark
Office in any notice of fee deficiency. 37 CFR 1.16td).
Filing Fee Calculation 35 932 ' 00

(Application Transmittal |4¥1|~page 4 of 7)

B. [:3 Design application
($280.00w37 CFR 1.160))
Filing Fee Calculation $
0, [:3 Plant applicatiOn
($460.00w87 CPR 1.165(9))

Filing fee calculation ‘ S
11. Small Entity Statement(s)
E] Verified Statementts) that this is a filing by a small entity under 37 CFR 1,9 and
127 islare) attached.
Filing Fee Calculation (50% of A, B or C above} 83
NOTE: Any excess of the full lee paid will be refunded if a verified statement and a refund request are filed within 2 months of the date of timely payment of a full fee. 37 OFF? i,28(a)i

t2. Request for International-Type Search (37 CPR 1.104(dli (complete, if applicable)

[3 Please prepare an international-type search report for this application at the
time when national examination on the merits takes 'place.
13. Fee Payment Being Made At This Time
[:1 Not Enclosed

[i No filing fee is to be paid at this time. (This and the surcharge required by
37 OFF? 1.16(e) can be paid subsequently.) - ‘
' Enclosed p
[g basicfilingfee ' $ 932,00
[:3 recording assignment ' I I
($40.00; 37 CPR 121(0)) 35
[j petition fee for filing by other
than all the inventors or person
on behalf of the inventor where
inventor refused to Sign or cannot
be reached. ($130.00; 37 CFR
1.47 and 1.1700) s

C] for processing an application with
a specification in a non~English
language. ($130.00; 37 CFR 152(0) and
l.17(k) $~Wm
[:1 processing and retention fee
($130.00; 37 CPR 153(0) and 1.210))
C] fee for international-type search report ($35.00;
37 CFR 1.21(e)). 15
NOTE: 37 CFR t.21(/) establishes a fee for processing and retaining any application which is abandoned for failing to complete the application pursuant to 37 CFH 7.53(d) and this, as‘weil as the changes to 37
CFR 153 and 1. 78, indicate that in order to obtain the benefit of a prior U. 8, application, either the ba-
sic iiiing fee must be paid or the processing and retention fee of § 1.21 (i) must be paid within 1 year
from notification under§ 53(d).
Total fees enclosed $ 922. 0 O

(Application Transmittal [4-11wpage 5 of 7)

14. Method of Payment of Fees
Check in the amount of 55W

{:1 Charge Account No.min the amount cl 3%. A duplicate of this transmittal is attached.
NOTE: Fees should be itemized in such a manner that it is clear for which purpose the less are pawl. 37 CPR
122(1)).

15. Authorization to Charge Additional Fees

WARNING: it no fees are to be paid on filing the following items should not be completed. ., WARNING: Accurately count claims, especially multiple dependent claims, to avoid unexpected high charges

it extra claim charges are authorized

The Commissioner is hereby authorized to charge the following additional fees

by1th1is paper) and during the entire pendency of this application to Account No

[Q 37 CFR 1.16(a). (t) or (g) (tiling tees)
[3 87 CFR 1.16(b), (c) and (d) (presentation of extra claims)
NOTE: Because additional tees for excess or multiple dependent claims not paid on filing or on later presenta‘
tion must only be paid or these claims cancelledby amendment prior to the expiration of the time peri-
od set for response by the PTO in any notice of tee deficiency (37 CFF? 1.16(d)), it might be best not to

authorize the PTO to charge additional claim lees except possibly when dealing with amendments at ter final action.

[Q 37 CFR 1.16(e) (surcharge for tiling the basic filing fee and/or declaration
on a date later than the filing date of the application)
[3 37 CFR 1.17 {application processing tees)
WARNING: While 37 CFR 1,17(a), (b), (c) and (d) deal with extensions 0/ time under § 1.136(a) this aulhoriza~ tion should be made only with the knowledge tnal “Submission of the appropriate extenSIon lee
under 37 C. F. Fl. 1 136(3) Is to no avail unless a request or petition (or extensionis tiled " (Empha
sis added) Notice of November 519851060(O. G 27).

[j 37 CFR 1.18 (issue tee at or before mailing of Notice of
Allowance. pursuant to 37 CPR 1.311(b))
NOTE: Where an authorization to charge the issue tee to-a deposit account has been filed before the mailing
of a Notice of Allowance, the issue lee will be automatically charged to the deposit account at the time
at mailing the notice of allowance. 37 CFF? 1.~311(b).
NOTE: 37 CFR 1.28(b) requires "Notification 01 any change in loss oi entitlement to small entity status must be
tiled in the application. prior to paying, or at the time of paying,.. issue lee“. From the wording at
37 OFF? 1 26(1)). (a) notification of change of status must be made even it the lee is paid as "other than
a small entity" and (b) no nolIlication is required it the change Is to another small entity.
16. instructions As To Overpayment
[:1 credit Account No.

refund . 74 S g

Reg. No. 3 3,1 3 O _ ‘ SIGNATURE OF ATTORNEY ‘ Christogher H. Gebhardt
Tel. NO. (41 2) 3 5 5-8 6 41 Type or print name of attorney

WKirkpatrick & Lockhart. P.O.Address 1500vOliver Building w”

‘ tsburgh, PA 15-222

(Application Transmittal [bu—page 6 cl 7)

[j Incorporation by reference of added pages

Check the following item if the application in this transmittal claims the ben-
efit of prior U.S. application(s) (including an international application enter-
ing the US. stage as a continuation, divisional or C-l-P application) and complete and attach the A0080 PAGES FOR NEW APPLICATION TRANSMITTAL WHERE BENEFIT OF PRIOR U.S. APPLICATIONtS) CLAIMED ‘

[3 Plus Added Pages For New Application Transmittal Where Benefit Of Prior US.
Application(s) Claimed
Number ol pages added

{:3 Plus Added Pages For Papers Referred To in item 4 Above
Number of pages added m

[:3 Plus "Assignment Cover Letter Accompanying New Application"
' Number ol pages added

Statement Where No Further Pages Added

(If no further pages form a part of this Transmittal then end this Transmittai with this page and check the following item) -

This transmittal ends with this page.

(Application Transmittal [4-115—page 7 of 7)

TITLE

SYSTEM AND METHOD FOR DISTRIBUTED STORAGE MANAGEMENTW
“““”””””””””

“ yam-W

NETWORKED ON COMPUTER 8Y§TEMS§'

Inventors:

James R. Woodhill
. Louis R. Woodhill
William Russell More, Jr.
Jay Harris Berlin

10

15

20

25

3O

BAG OUN OF THE.NVENTION

Field of the Igygptiop
The present invention is directed generally to a system
and method for distributed storage management on a networked computer system and, more specifically, to a system and method for distributed storage management on a networked computer i system including a remote backup file server and one or more local area networks in communication with the remote backup file server,

es ri o o the Ba k ton of the ve io

Backup copies of information stored on a computer system
must be made so that if a failure occurs which causes the original copies of the data to be lost, the lost data can be

recovered as it existed at the time when the last backup copy Ll was made. Backup/restore systems haVe a long history on‘all types of computer systems from mainframes to minicomputers,

local area netwOrk file servers and desktop workstations. I Historically, backup systems have operated by making

copies of a computer systemis files on a special backup input/output device such as a magnetic tape drive, floppy' diskette driVe, or optical disk drive. Most systems allow full backup, partial backup (e.g., specified drives, directories, or files), or incremental backups based on files changed after a certain date or time. Copies of files made during a backup rprocedure are stored on these special backup devices and are then later retrieved during a restore operation either under file names derived from the original file, from the date/time
of the backup operation or from a serially~incremented number. The backup procedure is typically'accomplished on an individual computer/file server basis, rather than through a single

“5,;

10

15

20

25

3O

the computer resources of two computers at most (the one processing the files to be backed up and the one with the backup device attached) are employed to effect the backup process, regardless of the actual number of computers networked together. V ‘ V
Today, the absolute numbers of computers networked
together by organizations are increasing rapidly as is the number of different types of computers and operating systems in use. At the same time, the number of storage deviCes and the capacities incorporated into each of these units is growing even more rapidly. in this environment, the backup/restore

‘ approaches which have been traditionally used have become less reliable, more expensive, and more consumptive of human time

and attention.

Thus, the need exists for a system designed to overcome
the limitations of the existing backup/restore systems that

have the following characteristics: (1) is capable of operating on a networked computer system incorporating various types of

computers and operating systems; (2) is capable of

accommodating a large array of large capacity storage devices;

(3) is reliable; (4) is capable of operating with a minimum: I amount of human intervention; and (5) is relatively

inexpensive.

sum/mm 92 111E; imagggg

The present invention is directed to a system for the distributed management of the storage space and data on a

networked computer system wherein the networked computer system includes at least two storage devices for storing data files comprised of one or more binary objects. The distributed

storage management system includes means for selectively

copying the binary objects stored on one of the storage devices

10

15

20

25

30

to another of the storage devices and means for calculating a current value for a binary object identifier.for selected binary objects stored on the storage devices wherein the

calculation of the binary object identifier is based upon the lactual data contents of the associated binary object. The

distributed storage management system further includes means
for storing the current value of the binary object identifier as a previous value of the binary object identifier, means for comparing the current value of the binary object identifier associated with a particular binary object to one or more

previous values of the binary object identifier associated vith; that particular binary object and means for commanding the

means for selectively copying binary objects in response to the means for comparing.

The present invention is further directed to a method for the management of the storage space and data on a computer system wherein the computer system includes at least two

storage area for storing data files cemprised of one or more

binary objects. The storage space management method includes the following steps: (1) Selectively copying the binary objects stored in one of the storage areas to another of the storage

areas; (2) calculating a current value for a binary object identifier for selected binary objects stored in the storage areas wherein the calculation of the binary object identifier
is based upon the actual data contents of the associated binary object: (3) storing the current value of the binary object identifier as a previous value of the binary object identifier; '(4) comparing the cUrrent value of.the,binary object identifier, associated with a particular binary object to one or more previous values of the binary object identifier associated with that particular binary object; and (5) controlling the step for

10

15

20

25

30

selectively copying binary objects in response to the step for

comparing.

' The system and method of the present invention for the management of the storage space on a computer system provide a backup/restore system that is capable of operating on a networked computer system incorporating various types of computers and operating systems, is capable of accommodating a large array of large capacity storage devices, is reliable, is capable of operating with a minimum amOunt of human

intervention and is relatively inexpensive. These and other

advantages and benefits of the present invention will become apparent from the description of a preferred embodiment

hereinbelow.

BRIEF DESCRIPTION OE 2H3 DgAWINGS
For the present invention to be clearly understood and

readily practiced, a preferred embodiment will now be described, by way of example only, with reference to the accompanying figures whereini V
FIG. 1 illustrates a simplified representation of a
networked computer system in which the system and method of the preseht invention may be employed;

FIG. 2 illustrates the manner in which the Distributed

Storage Manager program of the present invention allocates the storage space on each of the storage devices illustrated in

FIG. 1;
FIG. 3 illustrates the File Database utilized by the Distributed Storage Manager program of the present invention; :FIGI 4 illustrates the Backup Queue Database utilized by
the Distributed Storage Manager program of the present inventibn; and

10

15

20

25

30

‘FIGS. 5a — 51 illuStrate flow charts explaining the operation of the Distributed Storage Manager program of the present invention.

DETAILED DESCRIPTIO§ 0F TEE PREFfiRRED EMBODIQEEE
FIG. 1 illustrates a simplified representation of a
typical networked computer system 10 in which the system and method of the present invention for distributed storage management on networked computer systems may be employed. A remote backup file server 12 is in communication, via data path

13, with a wide area netWork 14. The wide area network 14 is, in turn, in communication with a plurality of local area

networks 16 via data paths 15. Those of ordinary skill in the

art will recognize that any number of wide area networks 14 may be in communication with remote backup file server 12 and that any number of local area networks 16 (from 1 to more than 100)
‘ may be in communication with each wide area network 14. Those of ordinary skill in the art will also recognize that the means

for communication between remote backup file server 12, wide

area network 14 and local area networks 16 over data paths 13
and 15 is well known.

Each local area network 16 includes multiple user workstations l8 and local computers 20 each in communication

with their respective local area network 16 via data paths 1?. Again, those of ordinary skill in the art will recognize that the means for communidation between user workstations 18, 1°°§;; computers 20 and local area networks 16 via data paths 17‘is well known. The storage space on each disk drive 19 on each localicomputer 20 in the networked computer system 10 is allocated as follows and as is shown in FIG. 2: (1) operating

system files 22; (2) a Distributed storage Manager program 24 which embodies the system and method of the present invention

10

15

20

25

3,0

(the operation of which is described in detail hereinbelow);
(3) a File Database 25 (the structure of which is described in detail hereinbelow); (4) a Badkup Queue Database 26 (the structure of which is described in detail hereinbelow); (5) local computer data files 28; (6) free disk space 30 and (7) compressed storage files 32 (created by the Distributed Storage Manager program 24 of the present invention as is explained

more fully hereinbelow). l l

The Distributed Storage Manager program 24 of the present invention builds and maintains the File Database 25 on one of the disk drives 19 on each local computer 20 in the networked

computer system 10 according to the structure illustrated in FIG. 3. The File Database 25 stores information relating to each file that has been backed up‘by the Distributed Storage

Manager program 24 since the initialization of that program on

each local computer 20. The File Database 25 is comprised of three levels of records organized according to a predefined

hierarchy. The top level record, File Identification Record

34,'includes identification information for each file that has been backed up by Distributed Storage Manager program 24. File Identification Record 34 contains the following elements: (1)

Record Type 36 (identifies the file as either a directory file

or a regular file); (2) File Location 38 (name of the directory in which the file resides); (3) File Name 40 (name of the

file); (4) Migration Status 41 (explained more fully

hereinbelow); and (5) Management Class 43 (explained more fully hereinbelow).

For each File Identification Record 34 in File Database

25, one or more Backup Instance Records 42 are created that contain information about the file (identified by File Identification Record 34) at the time that the file is backed

up. Each time that a file is backed up, a Backup Instance

10

15

20

25

30

Record 42 is created for that file. Each Backup Instance
Record 42 consists of the following elements: (1) Link to File
Identification Record 44; (2) Backup Cycle Identifier 46

(identifies the particular backup cycle during which the Backup Instance Record 42 is created); (3) Bile Size 48; (4) Last Modified Date/Time 50; (S) Last Access Date/Time 52; (6) File

Attributes 54 (e.g., read«only, system, hidden); (7) Delete
Date 56 (date on which the file was deleted); and (8) Insert

Date 57 (date on which the Backup Instance Record 42 was
, created).
Associated with each Backup Instance Record 42 is one or
more Binary Object Identification Records 58. The Distributed Storage Manager program 24 views a file as a collection of data
streams. A data stream is defined as a distinct collection of
data within the file that may be changed independently from
other distinct collections of data within the file. For

example, a file may contain its normal data and may also contain extended attribute data. A user may change the extended attribute data without modifying any of the normal data or vice versa; The Distributed Storage Manager program-24 further divides each data stream into one or more binary

objects. If the size of the data stream is equal to or less
than a previously defined convenient maximum binary object size (currently one (1) megabyte), then a single binary object
represents the data stream. If the data stream is larger than
the maximum binary object size, then the Distributed Storage Manager program 24 divides the data stream into multiple binary objects, all but the last of which are equal in size to the

maximum binary object size. A Binary Object Identification
Record 58 is created for each binary object that comprises the file which was backed up during the backup cycle identified by the Backup Cycle Identifier 46 of a particular Backup Instance

10

15

20

25

3O

Record 42. Each Binary Object Identification Record 58" includes the following components: (1) Link to Backup Instance Record 60; (2) Binary Object Stream_Type 62 (e.g., data,
extended attributes, security); (3) Binary Object Size 64; (4)

Binary Object CRC32 66 (explained more fully hereinbelow); (5) Binary Object LRC 68 (explained more fully hereinbelow): (6)

' Binary Object Hash 70 (explained more fully hereinbelow); and

(7) Binary Object Offset 72 (explained more fully hereinbelow).

The.Binary Object Size 64, Binary Object CRC32 66, Binary Object LRC 68 and Binary Object Hash 70 comprise the Binary Object Identifier 74 which is a unique identifier for each binary object to be backed up and is discussed in more detail

below.

The Distributed Storage Manager program 24 also builds and maintains the Backup Queue Database 26 on one of the disk
driyes 19 On each local computer 20 in the networked computer ‘1‘

system 10 according to the structure illustrated in FIG; 4.

Each entry (Backup Queue Record 75) in the Backup Queue Database 26 is comprised of the following components: (1)

Record Type 76 (identifies the file as either a directory file

or a regular file); (2) File Location 78 (name of the directory in which the file resides); (3) File Name 80 (name of the file); (4) File Status.82 ("new", "modified" or "deleted");l(5)

File Eize 84; (6) Lasthodified DateyTime 86; (7) Last Access Date/Time 88; (8) File Attributes 90 (e.g., readeonly, system, hidden); and (9) File Priority 92 (explained more fully hereinbelow).
The operation of the Distributed Storage Manager program

24 may be illustrated by way of the flow charts depicted in FIGS. 5a through Sl. For explanation purposes, the Distributed Storage Manager program 24 is divided into several distinct

i

'10

15

20

25

30

askill in the art will recognize, however, that each of the distinct functions operates in cooperation with the other 1 functions to form a unitary computer program.. Those of ordinary skill in the art will also recognize that the

following discussion illustrates the operation of the Distributed Storage Manager program 24 on a single local computer 20, although it should be understood that the

Distributed Storage Manager program 24 operates in the same‘

fashion on each local computer 20 on the networked computer

system 10. The Distributed storage Manager program 24 can:

either be executed on user demand or can be set to execute

periodically on a useredefined schedule.

1. Identification 9f Binary Objects to be gagged Up
In the flow chart of FIG. 5a, execution of the Distributed Storage Manager program 24 begins at step 100 where the Backup

QueueiDatabase 26 is built by creating a Backup Queue Record 75

for each File Identification Record 34 found in File Database
25. In this way, a list of files that were backed up during

the previous backup cycle is established so that it can be determined which files need to be backed up during the current backup cycle. To create each Backup Queue Record 75, the

Backup Instance Record 42 representing the most recent backup

of the file represented by each File Identification Record 34

is located. This determination is made by examining the Backup Cycle Identifier 46 in each Backup Instance Record 42. The Backup Cycle Identifier 46 may represent either a date '(month/day/year) or numerical Value assigned to a particular

backup cycle. The Backup Queue Record 75 is comprised of certain of the data fields of both the File Identification Record 34 and the Backup Instance Record 42. During.the

nY-nnoqq n‘f’l nvna+inn annh Rnnhnn (M15111: Dannvr‘! '7“; +‘ha wi'ln

10

15

20

25

3O

Status field 82 is set to "DELETED". However, if the Delete
Date field 56 of the most recent Backup Instance Record 42 associated with the File Identification Record 34 currently being processed is non—zero, indicating that the file has been previously deleted, then no Backup Queue Record 75 is created
for that File Identification Record 34. If the backup that is

currently being processed for the local computer 20 is not a full backup (i.e., all files on all disk drives 19 on the local

computer 20), then the Distributed Storage Manager program 24
' will only create Backup Queue Records 75 for those files that match the backup specificationsa For example, if only those files that have a file extension of ".EXE" are to be backed up, then only File Identification Records 34 that correspond to
".EXE" files will be processed.

Program control then continues with step 102 where the Distributed storage Manager program 24 of the present invention

scans all disk drives 19 on the local computer 20 that are to be backed up. This operation consists of scanning the
directory hierarchy on each disk drive 19 on the local computer ' 20 and returning to the Distributed Storage Manager program 24' certain file block information-for each of the directory files and regular files that are stored on the disk drives 19 to be backed up. A typical computer operating system maintains a file block for each file stored on the system which includes information such as file location, file type, user—assigned file name, file size, creation date and time, modify date and time, access date and time and file attributes. This operation may be controlled by some parameters that indicate which drives, directories and files are to be backed up during a

backup operation. However, the default operation is to back up all files on all disk drives 19 on the local computer 20.

Program control then continues with step 103 where the

10

15

20

25

3O

Distributed Storage Manager program 24 determines whether the

file block information for an additional file has been located

on the disk drives 19. If an additional file has been located,

program control continues with step 104. If an additional file

has not been located, program control continues with step 116. In step 104, the Distributed Storage Manager program 24 determines whether a Backup Queue Record 75 exists for the

located file by comparing the fileis file block information to the information stored in Backup Queue Database 26. If such a Backup Queue Record 75 does not exist ii.e.,'this is the first time this file will be backed up), program control continues with step 106 where a Backup Queue Record 75 for the-file is created using the information contained within the file‘s file block. The File Status field 82 for the newly created Backup Queue Record 75 is set to "NEW". Program control then continues with step 108 where a useredefined priority is assigned to the file and stored in the File Priority field 92 of the Backup Queue Record 75. This user—defined priority may be assigned to the file by methods that are well-known to those of ordinary skill in the art. The use of the File Priority field 92 by the Distributed Storage Manager program 24 is

discussed in more detail hereinbelow. ”Program control is then returned to step 102.
21f the Distributed Storage Manager program 24 determines,
in step 104, that a Backup Queue Record 75 exists in the Backup Queue Database 26 for the located file, program control continues with step 110 where it is determined whether any change has been made to the file. This determination is made

by comparing the information in the file's file block with the information stored in the file's Backup Queue Record 75. If i

any of the values have changed, program control continues with step 112 where File Status field 82 is set to "MODIFIED" and

10

15

20

25

30

the fields in the Backup Queue Record 75 are updated from the

file‘s file block information. Program control then continues

with step 108 where a user-defined priority is assigned to the

file and stored in File Briority field 92; program control is then returned to step 102. If the determination is made in step 110 that no change has been made to the file, then, in step 114, the Backup Queue Record 75 is deleted from the Backup

Queue Database 26 since the file does not need to be backed up.

BollOwing step 114, program control is returned to step 102.

If the Distributed Storage Manager program 24 determines,
, in step 103, that an additional file has not been located, program Control.continues with step 116. In step 116, the Distributed storage Manager program 24 reads each Backup Queue Record 75 in Backup Queue Database 26, one at a time. The Backup Queue Records 75 in Backup Queue Database 26 represent

all or the files that must be backed up by the Distributed' Storage Manager program 24 during the present backup cycle.

Program control continues with step 117 where the Distributed
Storage Manager program 24 determines whether a next Backup

Queue Record 75 has been located in Backup Queue Database 26.

If a next Backup Queue Record 75 has been located, program

control continues with step 118; otherwise, program control continues with step 119, where the routine illustrated by the flow chart of FIG. 5a is terminated. In step 118, the

Distributed Storage Manager program 24 determines whether the

File Status field 82 in the Backup Queue Record 75 currently being processed is set to "DELETED". If the File Status field 82 is set to "DELETED") pregram central continues with step 120 where the Delete Date field 56 in the most recent Backup Instance~Record 42 associated with the file identified by the Backup Queue Record 75 currently being processed is set toi

the

current date. A list of all Binary Object Identification

10

15.

20

25

30

Records 58 associated with the Backup Instance'Record 42 for the file identified by the Backup Queue Record 75 currently beinggprocessed is placed in a delete queue (not shown) that will be used by Distributed Storage Manager program 24 to
delete all Binary Object Identification Records 58 for binary

objects that have been deleted from the disk drives 19 of local computer 20. Program control then continues with step 122 where the Backup Queue Record 75 currently being processed is deleted from the Backup Queue Database 26. Program control is then returned to step 116.

If the Distributed Storage Manager program 24 determines,

in step 118, that the File Status field 82 of the Backup Queue Record 75 currently being processed is not set to "DELETED", program control continues with step 124 where the Distributed

storage Manager pregram 24 determines whether the File Status field 82 of the Backup Queue Record 75 currently being processed is set to "NEW". If the File Status field 82 is set ‘ to "NEW", program control continues with step 126 where a File

Identification Record 34 is created in File Database 25 using the information stored in the Backup Queue Record 75 currently
' being processed. Program control then continues with step 130. If the Distributed Storage Manager program 24 determines, in
step 124, that the File Status field 82 of the Backup Queue‘ Record 75 currently being processed is not set to "NEW" (i.e., the file has been modified since the last backup cycle),

program control continues with step 128 where the File Identification Record 34 associated with the file identified byli the Backup Queue Record 75 currently being processed is located in the File Database 25. Program control then continues with step 130: In step 130, the Distributed Storage Manager program

24 creates a new Backup Instance Record 42 in the File Database 25 for the file identified by the Backup Queue Record 75

10

15'

20

25

3O

.currently being processed. The Backup Instance Record 42 is
created using information stored in the associated File Identification Record 34 and the Backupqueue Record 75
currently being processed: The Backup Cycle Identifier 46 is
set to indicate that the file is to be backed up during the current backup cycle. The Delete Date field 56 is initialized

to "zero". The Insert Date field 57 is set to the current
date.

Program control then continues with step 132 where the Distributed Storage Manager program 24 separates the file

identified by the Backup Queue Record 75 currently being

processed into its component data streams. Each data stream is then processed individually. Those of ordinary skill in the art will recognize that these data streams may represent

regular datai extended attribute data, access control list

,data, etc.,Program control continues with step 134‘where the ‘Distributed Storage Manager program 24 determines whether each of the data streams currently being processed is larger than the maximum binary object size (currently one (1) megabyte).

If the data stream is larger than one (1) megabyte; program control continues with step 136 where the data stream currently being processed is segmented into multiple binary objects smaller in size than one (1) megabyte. Either following step 136 or, if the determination is made in step 134 that the data stream currently being processed is not larger than one (1)

megabyte (and, thus, the data stream is represented by a single binary object), program control continues with step 138.
In step 138, a Binary Object Identification Record 58 isbd created in File Database 25 for each of the binary objects

currently being processed. Each of these Binary Object

Identification Records 58 are associated with the Backup

Tread-ammo Damnvr‘l 11‘) nvaavkna in cud-on 1'20 mkm Dina-ax, fik4nn4‘

10

15

20

25

3O

Identifier 74 portion of each Binary Object Identification

Record 58 is comprised of the Binary Object Size field 64, the Binary Object ORC32 field 66, the Binary Object‘LRC field 68 and the Binary Object Hash field 70. Each of the fields of the Binary Object Identifier 74 may be four (4) bytes in length and
is calculated from the contents of each binary object. The
Binary Object size field 64 may be set equal to the byte—size,

of the binary object. The Binary Object CRC32 field 66 may be.

set equal to the standard 32—bit Cyclical Redundancy Check number calculated against the contents of the binary_object taken one (1) byte (8 bits) at a time. Those of ordinary skill in the art will readily recognize the manner in which the Cyclical Redundancy Check number is calculated. The Binary Object LRC field 68 may be set equal to the standard Longitudinal Redundancy Check number calculated against the contents of the binary object taken four (4) bytes (32 bits) at a time using the following algorithm:

BINARY OBJECT LRC = (initialized value)
for each double word (32 bits) of the binary object data:
LRC = LRC (XOR) double word of binary object data
end loop
The Binary Object Hash field 70 is calculated against the contents of the binary object taken one (1) word (16 bits) at a time using the following algorithm: \

HASH = (initialized value)
for each word (16 bits) of the binary object:
trotate current HASH value by 5 bits
iHASH == HASH + 1 .

‘HASH‘ HASH + (current word (16 bits) of binary object)
end loop I

i~

H

Since the Binary Object Identifier 74 is used to uniquely identify a particular binary object, it is important that the (possibility of two different binary objects being assigned the

10

15

29

25

3O

same Binary Object Identifier 74 be very small. This is the reason for implementing the Binary Object Identifier 74 using 128 bits and four separate calculations. Although a Binary
Object Identifier 74 may be calculated in various ways, the key notion is that the Binary Object Identifier is calculated from

the contents 0f the data instead of from an external and

arbitrary source. By incorporating the Binary Object Size field 64 within the Binary Object Identifier 74, only binary objects that are exactly the same size can generate duplicate‘ Binary Object Identifiers 74. Further, the calculations used' to determine-the Binary Object CRC32 field 66, the Binary Object LRC field 68 and the Binary object Hash field 70 are relatively independent of each other; Using the calculations set forth above, the probability that the Distributed Storage

Manager program 24 will generate the same Binary Object Identifier 74 for two different binary objects is extremely low.v Those of ordinary skill in the art will recognize that there exist many different ways of establishing the Binary Object Identifier 74 (e.g., establishing a Binary Object Identifier 74 of a different length or utilizing different .calculations) and that the procedure set forth above is only} one way of establishing the Binary ObjeCt Identifier 74.. The

critical feature to be recognized in creating a Binary Object

Identifier 74 is that the identifier should be based on the

contents of the binary object so that the Binary Object Identifier 74 changes when the contents of the binary object changes. In this way, duplicate binary objects, even if resident on different types of computers in a heterogeneous

network, can be recognized from their identical Binary Object Identifiers 74. 7

Program control then continues with step 140 where the Distributed Storage Manager program 24 identifies which binary"”

10'

15

20

25

3O

objects must be backed up during the current backup cycle. If

the File Status field 82 of the Backup Queue Record 75

currently being processed is set to "NEW", then all binary, objects associated with the file identified by the Backup Queue

Record 75 currently being processed must be backed up during
the current backup cycle. If the File Status field 82 is set to "MODIFIED", then only those binary objects associated withi the file that have changed must be backed upg Those binaryt objects that have changed are identified by comparing the Binary Object Identifiers 74 calculated in step 138 with the corresponding Binary Object Identifiers 74 associated with the next most recent Backup Instance Record 42 for the file identified by the Backup Queue Record 75 currently being procesSed. The Binary Object Identifiers 74 calculated in step 138 are compared against their counterparts in the File Database 25 (e.g.,-the Binary_object Identifier_74 (as calculated in step 138) that identifies the first binary object in the file (as determined by the Binary Object Stream Type fieldl62 and the Binary Object Offset field 72) is compared to

the Binary Object Identifier 74 (associated with the next most recent Backup Instance Record 42) for the first binary object in the file). This procedure allows the Distributed Storage

Manager program 24 to determine which parts of a file have

changed and only back up the changed data instead of'backing up all of the data associated with a file when only a small portion of the file has been modified. Program control is then

returned to step 116.

2. Concurrent QnsitelOffsite Backup
The Distributed Storage Manager program 24 performs two lconcurrent backup operations. In most cases, the Distributed Storage Manager program 24 stores a compressed copy of every

10

15

20

25

30

binary object it would need to restore every disk drive 19 on
every local computer 20 somewhere on the local area network 16 other than on the local computer 20 on which it normally resides. At the same time, the Distributed Storage Manager program 24 transmits every new or changed binary object to the 'remote backup file server 12. Binary objects that are available in compressed form on the local area network 16 can be restored very quickly while the much greater storage

capacity on the remote backup file server 12 ensures that at

leastfone copy of every binary object is stored and that a disaster that destroys an entire site would not destroy all copies of that site's data.
. The Concurrent Onsite/Offsite Backup routine begins at
step 200 of the flow chart illustrated in FIG. 5b where the Distributed Storage Manager program 24 compiles a list of those“ binary objects that are to be backed up during the current Vie backup cycle. Those binary objects which must be backed up during the current backup cycle are identified in step 140 of the flow chart of FIG. 5a. Those of ordinary skill in the art

will recognize, however, that the Concurrent Onsite/offsite

Backup routine may be performed independently of the routine

illustrated in FIG. 5a. ‘Program control then continues with step 202 where the Distributed Storage Manager program 24 identifies 1I whether there are any additional binary objects to be processed. If no additional binary objects are to be processed, program control is transferred to step 204 where the Concurrent Onsite/Offsite Backup routine is terminated. otherwise, program control continues with step 206 where the binary object currently being processed is compressed and stored in a compressed storage file 32 (FIG. 2) on one of the disk drives 19 on a local computer 20 on the local area network '1‘: nick-av- +hnn +415. lnna'l nnmnu+¢v on an ”kink +1143 hing-nu nh-ianh

10

15

20

25

3O

is currently stored. The compressed storage file 32 is used to allow the Distributed Storage Manager program 24 to pack several smaller compressed binary objects into a larger unit for storage. This is required to reduce the number of files

that the Distributed Storage Manager program 24 must manage and

to ensure that the Distributed Storage Manager program 24 does not create many "small" files since most file systems allocate

some minimum amount of Space to store a file even if the actual

file contains less data than the allocated space. The purpose behind storing the backup copy of a binary object on a disk

drive 19 on a different local computer 20 is to ensure that if
- the rirst disk drive 19 or local computer 20 fails, the backup copies othhe binary objects are not lost along with the original copies of the binary objects. .

Program control then continues with step 208 where each

compressed storage file 32, when it reaches a maximum

manageable size (e.g., two (2) megabytes), is transmitted to
the remote backup file server 12 (FIG. 1) over wide area network 14 for long—term storage and retrieval. Upon arriVal

of the compressed storage file 32 at the remote backup file

server 12, software resident on the remote backup file server ,12 routes the compressed storage file 32 for ultimate storage

to magnetic tape or other low cost storage media. The backupngl copy of a binary object which is maintained in the compressed storage file 32 on one of the disk drives 19 on one of the

local computers 20 is only the most recent version of each

binary object that is backed up while the backup copy of the binary object stored on the remote backup file server 12 is kept until it is no longer needed. Since most restores of

file; on a local area network 16 consist of requests to restore

the most recent backup version of a file, the local copies of binary objects serve to handle very fast restores for most

10

15

20

25

30

restore requests that occur on the local area netWork 16. If. the local backup copy of a file does not exist or a prior versidn of a file is required, it must be restored from the
remote backup file server 12. Program contro1 then continues with step 210 where the distributed Storage Manager program 24

determines whether sufficient space is available in the space

allocated for compressed storage files 32 on the disk drives 19 on local computers 20 for storage of the binary object currently being processed. If sufficient space is available,

program control is returned to step 200. 'Otherwise, the binary object currently being processed is deleted from the disk drive

19 on which it was stored after transmission‘to the remote

backup file server 12 has been completed. Program control is then returned to step 200.

3. File Prioritization

The file prioritization process performed by the Distributed Manager Storage program 24 is handled by four
1 .

interrelated routines of that program: (1) Backup/Restore
1 . .

Routine; (2) Compression Routine; (3) Local Storage Routine;
, ‘ ,

and (4) Resource Allocation Routine. Each routine will be
i described in turn. In the following discuSsion, when one of

the four routines is discussed, it should be understoOd that it is the Distributed Storage Manager program 24 that is executing

the functions of that routine. The Backup/Restore Routine, the
Local Storage Routine and the Compressioanoutine may be executed on each of the local computers 20 on the networked computer system 10 while the Resource Allocation Routine is executed on only one of the local computers 20 on the networked computer System 10. This execution scheme permits the resources of any available local computer 20 on any of the

local area networks 16 to be utilized according to its,

10

15

20

25

3O

availability. Furthermore, more than one local computer 20 may be utilized to complete any high-priority tasks required to be

completed within a specified time frame. An advantage of the process of prioritization of files is that it allows the Distributed Storage Manager program 24 to effectively deal with

a situation where local storage and wide area network t transmission resources are limited. The Distributed Storage Manager program 24 is also able to keep track of data which is not stored locally or transmitted to the remote backup file

server 12 in any giVen backup cycle and then attempt to resume

these processes during the next backup cycle.
The Backup/Restore Routine is illustrated in the flow.
chart shown in FIG. 5c.‘ In step 300, the Distributed Storage

Manager program 24 initiates the Backup/Restore Routine by

locating the highest priority binary object scheduled for

backup on the local computer 20 on which the Backup/Restore

Routine is executing. The identities of the binary objects to be backed up and their respective priorities were determined by

the Distributed storage Manager program 24 in the flow chart of FIG. 5a. Those of ordinary skill in the art will recognize,

rhowever, that the file prioritization routines of the

Distributed Storage Manager program 24 may be utilized independently of the process illustrated in the flow chart of‘ FIG. 5a. Program control then continues with step 302 where the Backup/Restore Routine of the Distributed Storage Manager program 24 determines whether a binary object to be backed up has been located. If not, program control continues with step 304 where the Backup/Restore Routine is terminated. Otherwise,
, program control continues with step 306 where the Backup/Restore Routine of the Distributed Storage Manager program 24 sends a message to the Resource Allocation Routine indicating the priority of the highest priority.binary object

10

15

2O

25

30

that the Backup/Restore Routine has located. Program control
then continues with step 308 where the Backup/Restore Routine I waits for a_message from the Resource Allocation Routine indicating which Compression Routine is available to compress and store the highest priority binary object located by the Backup/Restore Routine. In this way, the Distributed Storage Manager program 24 is able to perform not only local computer

26 based file prioritization but also networked computer system ' 10 based file prioritization. This is accomplished_by having the Resource Allocation Routine examine the priority of the

highest priority binary object located by each Backup/Restore

Routine and then allocating compression resources to the

Backup/Restore Routine which has the highest priority binary object to compress. ’ ‘

Program control continues with step 310 where the

Backup/Restore Routine receives a message from the Resource Allocation Routine indicating that a Compression Routine is

available for binary object compression, The Backup/Restore

Routine then sends a list of up to forty (40) binary objects or up to one (1) megabyte of uncompressed binary objects to the

Compression Routine starting with the highest priority binary object that the Backup/Restore routine has identified for backup. The reason to limit the number or size of binary objects that are sent to a Compression Routine is to allow the

Compression-Routine to work for a limited amount of time on the binary objects for one Backup/Restore Routine before becoming available to work on another Backup/Restore Routine's binary objects.
The Compression Routine performed by the Distributed

Storage manager program 24 is illustrated in the flow chart depicted in FIG. 5d. Program control begins at step 312 where the Compression Routine of the Distributed Storage Manager

10

‘15

20

25

30

program 24 sends a message to the Resource Allocation routine

indicating that the Compression Routine is available to compress binary objects. Program control then continues with step 314 where the Compression Routine waits for a compress message from a Backup/Restore Routine indicating which binary objects are to be compreSsed. The compress message includes

the File Name 40 from Identification Record 34, the Binary Object Stream Type field 62 from the Binary Object

Identification Record 58 and the Binary Object Offset field 72 from the Binary Object Identification Record 58 for each binary object to be compressed. The binary objects are placed in a “compression queue (not shown) for processing by the Compression Routine.‘ Program control then continues with step 316 where the Compression Routine sends a message to the Resource

Allocation Routine to determine which Local Storage Routine has

space available for storage of compressed binary objects. Program control then continues with step 318 where the Compression Routine requests allocation of.a compressed storage file 32 from the Local Storage Routine that has indicated

availability of storage space. Program control continues with

step 320 where the binary object is compressed and stored in the allocated compressed storage file 32. JPrcgram control then continues with step 322 where the Compression Routine

determines whether there are more binary objects in the compress queue. if there are no more binary objects present in the compress queue, program control returns to step 312. If more binary objects are present, program control continues with step 324 where the Compression Routine determines whether the allocated compressed storage file 32 is full. If not, program control is returned to step 320. Otherwise, program control is

returmed to step 316 where the Compression Routine sends a

message to the Resource Allocation Routine to determine which

10

15

20

25

30

Local Storage Routine has spaCe available for storage of

compressed binary objects.

The Local Storage Routine executed by the Distributed
Storage Manager program 24 is illustrated in the flow chart

depicted in FIG. Be. The Local Storage Routine is responsible for managing storage file space on a particular local computer
20. Rrogram control beings at step 326 where the Local Storage

Routine of Distributed Storage Manager program 24 sends a message to the Resource Allocation Routine indicating the amount of storage space it has available for allocation of

compressed storage files 32. The Local Storage Routine

determines the amount of space it has available for allocation of compressed storage files 32 by determining the total amount 'of free space on its disk drives 19 and then determining how must space must be left as "free space". The amount of required Pfree space" is user~specified. Program control

continues with‘step_328lwhere the Local Storage Routine waits for a request from a Compression Routine for allocation of a compressed storage file 32. Upon receipt of such a request,
‘ program control continues with step 330 where the requested
}

compressed storage file 32 (e.g., two (2) megabytes in size) is allocated. The Local Storage Routine then returns a message to'

the requesting Compression Routine indicating the name and location of the compressed storage file 32 that has been allocated. Program control is then returned to step 326.
The Resource Allocation Routine performed by the

Distributed Storage Manager program 24 of the present invention is depicted in the flow chart of FIG. 5f. The Resource Allocation Routine is a process that responds to messages fromifl‘ other routines of the Distributed Storage Manager program 54
and allocates resources between resource reguesters and
resource providers. Program control begins with step 332 where

10

15

20

25

. 30

the Resource Allocation Routine executed by the Distributed Storage Manager program 24 waits for a message from a " '

Distributed Storage Manager program 24 routine. When a message
is received, program control continues with step 334 where the‘fiw Resource Allocation Routine determines whether the message is

from a Backup/Restore Routine transmitting information relating to its highest priority binary object for compression. If such

a message is received, program control continues with step 336 whereithe Resource Allocation Routine stores this information
in an internal table containing Backup/Restore Routine status

information. The Resource Allocation Routine then scans this

status information table to ascertain which Backup/Restore Routine has the highest priority binary object for storage. Program control then continues with step 338 where the Resource Allocation Routine determines whether any Compression Routine

is available to process the highest priority binary object. If no Compression Routine is available for processing, program control is returned to step 332. If an available Compression Routine is located, program control continues with step 340 where the Resource Allocation Routine transmits a message to

the requesting Backup/Restore Routine indicating which

Compression Routine is available to compress the binary object.)

In addition, the Resource Allocation Routine marks the

Compression Routine as_"working" in an internal table containing Compression Routine information. 'Program control is

then returned to step 332.

- If the Resource Allocation Routine determines, in step

334, that the received message is not from a Backup/Restore Routine, program control continues with step 342 where the

Resource Allocation Routine determines whether the received‘

message is from a Compression Routine indicating that the transmitting CompreSsion Routine is available for processing.

10

15

20

25

'30

,If the received message is from a Compression Routine, program control continues with step 344 where the Resource Allocation

Routine marks the transmitting Compression Routine as‘

“available" in its internal table containing Compression

Routine information. Program control then continues with step

336.

If the Resource Allocation Routine determines, in step
342, that the received message has not been transmitted from a Compression Routine indicating its availability for processing, program control continues with step 346 where the Resource

Allocation Routine determines whether the received message is from a Local Storage Routine indicating the amount of storage

space that the Local Storage Routine has available. If the

received meSSage is from a Local Storage Routine, Program control continues with step 348 where the Resource Allocation Routine locates the transmitting Local Storage Routine in an internal table containing Local Storage Routine information and

saves the storage space information transmitted by the Local Storage Routine. Program control then continues with step 354. If the Resource Allocation Routine determines, in step 346, that the received message is not from a Local Storage Routine, program control continues with step 350 where the Resource Allocation Routine determines whether the received message is from a Compression Routine requesting a compressed storage file
32. If the Resource Allocation Routine determines that such a message was received, program control continues with step 352

where the identity of the requesting Compression Routine is added to a "space request list". ‘Rrogram control then continues with step 354; If the Resource Allocation Routine determines, in step 350, that the received message is not from

a Compression Routine requesting a compressed storage file 32,

10

15

20

25

3O

In step 354, the Resource Allocation Routine determines
whether any Compression Routines are waiting for allocation of

a compressed storage file 32 by examining the "space request
list",i If no such Compression Routines are in the "space request list", program control is returned to step 332. If the identity of such a Compression Routine is found in the "space

request list“, program control continues with step 356 where
the Resource Allocation Routine determines whether any of the

'Local Storage Rrocessors has any available space by examining
- the information in its internal table containing Local Storage

Routine information. When a Compression Routine requests a compressed storage file 32, the Compression Routine identifies, the name of the local computer 20 on which the Backup/Restore Routine is executing and on whose behalf it is compressing
binary objects. This allows the Compression Routine to request
a compressed storage file 32 on a local computer 20 other than the local computer 20 on which the binary object to be stored

resides. Otherwise, the backup copy of the binary object may be sthred on the same local computer 20 as the original binary ‘. object whereby a disk drive failure would result in losing bothw the original and backup copies. The Resource Allocation

Routine uses the information supplied by the Compression

Routihe to'ensure that the requested compressed storage file 52 is allocated on a local computer 20 other than the local

computer 20 from which the binary object originated. If available storage space is located, program control continues with step 358 where the Resource Allocation Routine transmits a message to the next Compression Routine in the "space request list" indicating which Local Storage Routine has allocated an

available compressed stOrage file 32. Program control is then returned to step 354.

10

15

20

25

3O

If the Resource Allocation Routine determines, in step

356, that no storage space is available, program control

continues with step 360 where the Resource Allocation Routine determines whether there are any compressed storage files 32

that are maintained by any of the Local Storage Routines which have A lower priority that the binary object currently being

processed. If so, program control continues with step 362 whereia message is transmitted to the Local Storage Routines instructing the Local Storage Routines to delete some of the low-priority compressed storage files 32 to make room for 'higher priority binary objects. After these lower—priority compressed storage files 32 are deleted by the Local Storage

Routines, the Local Stdrage Routines will transmit new status messages to the Resource Allocation Routine. Program control is then returned to step 332. If no lower«priority compressed storage files 32 are located in step 360, program control
. continues with step 364 where the Resource Allocation Routine transmits a message to the Local Storage Routines with

instructions that from that time forward, any allocated compressed storage files 32 are to be deleted after the

contents of the compressed storage files 32 have been successfully transmitted to the remote backup file server 12 for long-term storage. Program control is then returned to

step 332.

4. granularizgtion o; Eileg
The most important class of "large" files on computer
systems such as networked computer system 10 is databases.

TypiCallyy on a given day, only a small percentage of the data in a large database is changed by the users of that database. However, it is likely that some data will be changed in each one of the (1) megabyte binary object segments that are created

10

15

20

25

30

in step 136 of the flow chart depicted in FIG, 5a. As a

result, in most cases, the entire "large" database file would

have to be backed up to the remote backup file server 12;

However, the Distributed storage Manager program 24 of the present invention utilizes a technique of subdividing large database files into "granules" and then tracks changes from the previous backup copy at the "granule" level. The "granule" size utilized by the Distributed Storage Manager program 24 may

be one (1) kilobyte although those of ordinary skill in the art

will recognize that any "granule" size that produces the most

efficient results (in terms of processing time and amount of data that must be backed up) may be utilized. This technique of subdividing files into "granules" is only used to reduce the amount of data that must be transmitted to the remote backup file server 12 and is not utilized in making backup copies of

binary objects for storage on local computers 20.

The operation of the Distributed Storage Manager program
24 in subdividing files into "granules" is illustrated in the

flow chart depicted in FIG. 5g.. This "granularization" procedure is performed for "large" files following step 136 of the flow chart of FIG. 5a. Program control begins at step‘goo where the Distributed Storage Manager program 24 identifies whether the binary object currently being processed is.a segment of‘a "large" database-like file. Program control then continues with step 402 where the Distributed Storage Manager program 24 determines whether this is the first time that the

binary object currently being processed is being backed up using the "granularization" technique. If so, program control continues‘with step 404 where the Distributed Storage Manager program 24 creates a "shadow file" which contains a "contents identifier" for each "granule" in the binary object currently being processed. Each "contents identifier" is composed of a

10

15

20

25

30

standard 32%bit Cyclical Redundancy Check number which is ‘calcuflated against the contents of the “granule" and a 32-bit hash number which is calculated against the contents of the "granule" in the same manner described in relation to step 138 of the flow chart depicted in FIG. 5a. Those of ordinary skill in the art will readily recognize the manner in which the Cyclical Redundancy Check number is calculated. Each time that

the binary object is to be backed Up, the Distributed Storage

Manager program 24 can calculate the "contents identifier" for each "granule" in the binary object and then compare-it to the "contents identifier" of the "granule" the last time the binary‘p

object was backed up and determine if the ”granule" hast

changed. This allows the Distributed Storage Manager program
24 to determine what data within a binary object has changed
and only back up the changed data instead of the entire binary object. Program control then continues with step 406 where the Distributed Storage Manager program 24 calculates a "change

identifier" for each "granule" of the binary object and stores‘

it inithe "shadow file" for that binary object. Program control then continues with step 408 where the binary object is

compressed into a compressed storage file 32 which becomes the most recent complete copy of the binary object for later reconstitution of the binary object as is discussed more-fully hereinbelow. The contents of the compressed storage file 32 is

then transmitted to the remote backup file server 12 for long~ term storage and retrieval. Program control is then returned to step 400.

If the Distributed Storage Manager program 24 determines,
in step 402, that this is not the first time that the binary object currently being processed is being backed up using the

"granularization" technique, program control continues with step 410 where the Distributed Storage Manager program 24

10

15

20

25

3O

calculates the "contents identifier" for each "granule"}‘

Program control continues with step 412 where each newly— calculated "contents identifier" is compared to the

corresponding "contents identifier" for the "granule" in the "shadow file". If the two values are equal, program control centinues with step 414 where the Distributed Storage Manager progrAm 24 determines whether the last "granule" of the binary

object has been processede If soi program control is returned ‘to step 400; otherwise, program control continues at step 410.

If the "contents identifiers" are not found to be equal in step

412, the "granule" has changed and program control continues with step 416 where the "shadow file" is updated with the newly~ca1culated "contents identifier" for the "granule". Program control then continues with step 418 where the changed "granule" is compressed into a compressed storage file 32 using a special format that identifies the "granule". All changed "granules" for the "data stream" currently being processed are packed together in the same compressed storage file.32. The contents of the compressed storage file 32 is then transmitted)” to the remote backup file server 12 for longeterm storage and retrieval.‘ If the Distributed Storage Manager program 24 determines that a large percentage of the "granules" in the

binary object have changed (e.g., 80%), then the entire binary object is backed up to the remote backup file server l2.

The operation of the Distributed Storage Manager program
24 in reconstituting, on a local computer 20, a binary object that has been transmitted to the remote backup file server 12 using the "granularization" technique illustrated in FIG. 5g is. illustrated in the flow chart depicted in FIG. 5h. Program control begins at step 420 where the Distributed Storage

Manager program 24 creates a work area on the remote backup
file server 12 that is equal in size to the total uncompressed

10

15

20

25

30

size of the binary object that is to be reconstituted. "Program control continues with step 422 where the most recent complete

copy of the binary object to be reconstituted is located on the remote backup file server.12 and is decompressed into the work

area. Program control then continues with step 424 where the Distributed Storage Manager program 24 creates a bitmap with

one bit representing each granule of the binary object to be

reconstituted. Initially, all bits in this bitmap are set to zero (0). Each bit in the bitmap is used to indicate whether

the granule associated with that bit has been restored to the

most recent complete copy of the binary object. Program controd then continues with step 426 where the Distributed Storage Manager program 24 locates the most recent "granularized" copy of the binary object that was stored on the

remote backup file server 12. Each time that step 426 is executed, the next most recent “granularizedfi copy of the ‘binary object is located] This process continues until all

bits in the bitmap are set_to one (1) or until there are no more "granularized" copies of the binary object that are newer than the most recent complete copy of the binary object. At that point, the binary object will have been reconstituted and will be ready to be restored to the local computer 20. Following step 426, program control continues with step 428 where the Distributed storage Manager program 24 determines

whether another "granularized" copy of the binary object has

been lbcated. If so, program control continues with step 430 where the Distributed Storage Manager program 24 obtains’the»

list of "granules" in the "granularized" copy of the binary object just located. If another “granularized” copy of the

binary object is not located in step 428; program control continues with step 438 where the reconstituted binary object

is restored to the local computer 20. Following step 430,

10

15'

20

25

30

program control continues with step 432 where, starting with

the first "granule" in the "granularized" copy of the binary object, the Distributed Storage Manager program 24 determines 2' whether the bit for this "granule" in the bit map is set to zero (0). If the bit is set to one (1), a more recent copy of

the "granule" has already been decompressed and copied into the
work area. If the bit is set to Zero (0), program control
continues with step 434 where the "granule" is decompressed and copied into the work area at the correct location for that
"granule". After copying the "granule" to the work area, the'
Distributed Storage Manager prbgram 24 sets the bit within the

bitmap for the "granule" to one (1). If the Distributed Storage

Manager program 24 determines, in step 432, that the bit is not set to zero (0), program control continues with step 440 where the Distributed Storage Manager program 24 determines whether
' therejare any more "granules" to be processed in the current

set 0% “granules". If so, program control is returned to step

432; otherwise, program control is transferred to step 426. Following step 434, program control continues with step 436 where the Distributed Storage Manager-program 24 determines whether all bits in the bitmap are now set to one (1). If so,

program control continues with step 438 where the reconstituted

binary object is restored to the local computer 20. If the
Distributed Storage Manager program 24 determines, in step 436,
that all bits in the bitmap are not set to one (1), program i control continues with step 440.
The technique of "granularizing" "large" files also
becomes useful when a current version of a file (comprised of current versions of binary objects) must be restored to a

previous version of that file (comprised of previous versions of binary objects). Each binary object comprising the current version of the file can be restored to the binary object

10

15

2O

25

30

comprising the previous version of the file by restoring and updating only those "granules" of the current version of the binary objects that are different between the current and

previous versions of the-binary objects.t This technique is illustrated in the flow chart depicted in FIG. 5i. Program

control begins at step 442 where the Distributed Storage
Manager program 24 obtains from the user the identities of the current and previous versions of the file (comprised of binary objects) which needs to be restored. Program control continuese

with step 443 where the Distributed Storage Manager program 24 compiles a list of all binary objects comprising the current version of the userespecified file. This information is

obtained from File Database 25. Program control then continues with step 444 where the Distributed Storage Manager program 24

calculates "contents identifiers" for each "granule" within the current version of each binary object as it exists on the local computer 20. Program control then continues with step 446 where the Distributed Storage Manager program 448 transmits an "update request" to the remote backup file server 12 which

includes the Binary Object Identification Record 58 for the previous version of each binary object as well as the list of "contents identifiers" calculated in step 444. Program control continues with step 448 where the Distributed Storage Manager

program 24 reconstitutes each previous version of the binary objects according to the technique illustrated in the flow chart depicted in FIG. 5h. Program control then continues with step 450 where the Distributed Storage Manager program 24, for each binary object, compares the "contents identifier" of the next "granule" in the work area of remote backup file server 12 Vagainst the correspdnding "contents identifier" calculated in Step 444. ~Program control continues with step 452 where the
x Distributed Storage Manager program 24 determines whether the

10

15

20

25

3O

' computer 20. Program control then continues with step 456

"contents identifiers" match. If so, program control is returned to step 450 since this "granule"-is the same on the

local computer 20 and on the remote backup file server 12. If-the Distributed Storage Manager program 24 determines, in step 452,-that the "contents identifiers" do not match, program control continues with step 454 where the Distributed Storage Manager program 24 transmits the "granule" to the local

whereéthe "granule" received by the local computer 20 is written directly to the current Version of the binary object at the appropriate location. Program control then continues with step 458 where the Distributed Storage Manager program 24 determines whether there are any more "granules" to be examined for the binary object currently‘being processed. If so,

program control is returned to step 450} otherwise the file

restore routine is terminated at step 460. After all

"granules" are received from the remote backup file server lz,aji the binary object has been restored to the state of the' previous version.

5. Auditing and Reporting
The Distributed Storage Manager program 24 is able to
perform self~audits on a periodic basis to ensure that the binary objects that have been backed up can be restored. To perform an audit, the Distributed Storage Manager program 24

executes the steps illustrated in the flow chart of FIG. Sj. Program control begins at step 500 where the Distributed Storage Manager program 24 initiates a restore of a randomly selected binary object identified by a Binary Object Identification Record 58 stored in File Database 25. Program control continues with step 502 where the selected binary obiect is restored from either a compressed storage file 32

10

15

20

25

3O

residing on one of the disk drives 19 of one of the local computers 20 or from the remote backup file server 12. ‘Program control then continues with step 504 where, as the binary object is being restored, a Binary Object Identifier 74 is ‘calculated from the binary object instead of writing the binary object to one of the disk drives 19 of one of the local computers 20. Program control then continues with step 506 where the Distributed Storage Manager program 24 compares the Binary Object Identifier 74 calculated in step 504 to the

original Binary Object Identifier 74 stored as part of the randomly selected Binary Object Identification Record 58. If the values are equal, program control continues with step 508 where the Distributed Storage Manager program 24 logs a

successful audit restore. If the values are not equal, program " x ‘contrbl continues with step 510 where the Distributed Storage Manager program 24 generates an event indicating an audit
failure.

6. Virtual Restore

The disk drives 19 associated with local computers 20 may
have a very large storage capacity and may require a
significant amount of time to be restored, especially if most‘ or all of the data must be transmitted from the remote backup file server 12. To reduce the amount of time that a local computer 20 is "offline" during a full disk drive 19 restore, the Distributed Storage Manager program 24 employs a technique;‘, which allows a disk drive 19 associated with a local computer
20 to be only partially restbred before being put back "online" for access by local computer 20. The user specifies to the

Distributed Storage Managerhprogram 24 that only those files

thatghave been accessed in the last <n> days, <n> weeks or <n>

months should be restored to the disk drive 19 before the disk

10

15

20

25

30

drive 19 is returned to the "online“ state. Alternately, the,,f user may specify that only files that are stored "loCally" in compressed storage files 32 should be restored and that no
files stored on the remote backup file server 12 should be restored before the disk‘drive 19 is returned to the "online"

state. The overall result is a minimization of restore time in the event of disk drive 19 failure.' This "virtual restore“

technique generally works quite well since users who will begin

accessing data on'a particular disk drive 19 after it is put

back "online" will most likely only be accessing data that had been "recently" accessed before failure of the disk drive 19. The "virtual restore" process is illustrated in the flow
chart depicted in FIG. 5k. Program control begins at step 600 where the Distributed Storage Manager program 24 obtains, from the user, the last access date that defines which files must be

restored before the disk drive 19 can be returned to the

“online” condition. Any files that were last accessed on or aftergthis date will be restored before the disk drive 19 is placed "online". The specification of this date may be aocomfilished in any of the following ways:/ (1) actual date;
(2) "within the last <n> days"; (3) "within the last <n>

weeks"; or (4) "within the last <n> months". Alternately, the user may specify that only files that are currently backed uh ,in compressed storage files 32 are to be restored as opposed to files stored on remote backup file server 12. Program control

continues with step 602 where the Distributed Storage Manager

Program 24 locates the most recent version of the File Database 25 for the disk drive 19 to be restored if the File Database 25 does not already exist on the disk drive 19. Program control then continues with step 604 where the next File Identification Record 34 in File Database 25 is read. 'Program control continues with step 606 where the Distributed Storage Manager

10

15

20

25

3O

program 24 determines whether an additional File Identificationi

Record 34 to be processed has been located in File Database 25.

If not, program control continues with step 608 where the Distributed Storage Manager program 24 notifies the local computer 20 that the restored disk drive 19 may be placed "online" and terminates the virtual restore process. If

another File Identification Record 34 has been located for

processing, program control continues with step 610 where the4 Distributed Storage Manager program 24 locates the most recentMM

Backup Instance Record 42 associated with the File Identification Record 34 currently being processed. In step 612, the Distributed Storage Manager program 24 determines

whether the Last Access Date/Time field 52 in the Backup

Instance Record 52 indicates that the file has been accessed

since the user~specified last access date (step 600). If the

file has been accessed on or since the usersspecified last access date, program control continues with step 614 where the Distributed Storage Manager program 24 initiates the

restoration of this file and sets the Migration Status field 41

in the File Identification Record 34 currently being processed to "NORMAL". Program control is then returned to step 604; If the Distributed Storage Manager program 24 determines, in step

612, that the file has not been accessed on or since the user—

specified last access date, program control continues With step
616 where the Distributed Storage Manager program 24 sets the Migration Status field 41 in the File.Identification Record 34 currently being processed to "MIGRATED". In this case, the file does not need to be restored. Program control is then

returned to step 604;
Another feature of the virtual restore precess is the

ability to Utilize the Migration Status field 41 in File, 'Identification Record 34 for the performance of space‘

10

15

20

25

30

managementl If a particular file has not been accessed on or
since a user—specified last access date, the file can be backed

up to the remote backup file server 12 and then deleted from

the disk drives 19 associated with local computers 20. ‘The Migration Status field 41 is then set to "MIGRATED". If a migrated file is later needed by,a user, the file can be

restored from the remote backup file server 12.

7. Backup File getention

The Distributed Storage Manager prcgram 24 implements a
backup file retention scheme wherein a retention pattern is maintained for each individual file that indicates which‘backup versions of a file are to be saved. A retention pattern for a

file is defined as:

keep the last "d" daily backup copies of the file AND
keep the last "w" weekly backup copies of the file AND
keep the last "m" monthly backup copies of the file AND.
keep the last "q" quarterly backup copies of the file AND
keep the last "y" yearly backup copies of the file.

By specifying the retention pattern in this way, all backup copies of a file that are needed to represent the backup of the file as it existed at the time it was backed up for the last "d" days, the last "w" weeks, the last "m" months, the last "q" quarters and the last "y" years are saved. However, by way of example, this may mean that only one backup cOpy of the file is

saved to represent all "d" days (in the case where the file has not changed in the last "d" days), or this may mean that the last "d" daily backup copies of the file must be saved to

represent the file as it existed for the last "d" daily backup cycles. The same principle is utilized for weekly, monthly,

quarterly and yearly copies.

i

l

10

15

20

25

,30

The backup file retention scheme utilized by the

Distributed Storage Manager program 24 provides several unique benefits. First, this technique prevents undetected virus or

application program damage to‘a file from destroying all good backup copies of a file. If a file is damaged and this condition is not noticed for several days, then a scheme which only maintains the last "n" versions of a file may result in the situation where an "undamaged" backup copy of the file is

not available. The backup file retention scheme of the present invention allows backup copies of files to be kept that represent the file as it existed at various times during the

past several days, weeks, months or even years; Second, the

file retention scheme utilized by the Distributed Storage
. Manager program 24 eliminates the need for most archives. Most archives are designed to take a snapshot of a group of files as
of a certain date, such as at the end of each month. The Distributed Storage Manager prbgram's use of retention patterns

eliminates the need for users to take periodic snapshots of

their data using a special archive, since the Distributed Storage Manager program 24 handles this automatically.
In order for the Distributed storage Manager program 24 to implement the backup file retention scheme, each file stored on the local computers 20 must be associated with a specific

retention pattern. The Management Class field 43 in the File Identification Record 34 of File Database 25 specifies a management class for each file. In turn, each management class is associated with a specific file retention pattern. In this
0 way, a specific retention pattern is associated with each file. Those of ordinary skill in the art will recognize that other methods of aesigning a specific file retention pattern to a file may also be utilized.

10

15

'20'

25

3O

9

The operation of the backup file retention scheme utilized
by the Distributed Storage Manager program 24 is illustrated in the flow chart of FIG. 51. Program control begins at step 700 where the Distributed Storage Manager program 24 locates each

File Identification Record 34 in the File Database 25.. Program

contrdl continues at step 702 where the Distributed Storage

Manager program 24 determines the required file retention pattern by examining the Management Class field 43 in the File Identification Record 34 currently being processed and then creates a "retention working list". The ”retention working list" is a list of entries that specify the starting and ending dates for each backup copy that should be retained based upon the specified retention pattern. For example, if the user has specified that the last "d" daily backup copies must be retained, then the "retention working list" will contain "d"

entries with the "start date" equal to the "end date" for each entry and the dates for the-first entries set equal to the current date, the dates for the second entries set equal to the previous dayfs date, etc. For weekly entries, the "retention working list" will contain entries (one per weekly backup copy

to beéretained) with the “start date" set to the date that
I specifies the beginning of the prior week (based on the current date)?and the "end date" set to the date that specifies the end of the prior week (based on the current date). If "w" weeks

are to be retained, then "w" weekly "retention working list" entries will be created. At the end of this process, the

"retention working list" will contain a list of "windows" which

indicates the date ranges that a file must fall within in order to be retained. ‘

Program control continues with step 704 where the Distributed Storage Manager program 24 locates the most recent

nanknh Thc+2hnn«nannvfl A? scannia+ad mi+h +Hb W41o

10

15

20

25

30

Identification Record 34 currently being processed. ‘Program

control continues with step 706 where the Distributed Storage Manager program 24 compares the date stored in the Insert Date field 57 of the Backup Instance Record 42 currently being

processed with any "unused" date ranges set forth in the

"retention working list" (if any of the‘firetention working
list" entries have already been satisfied, they will be marked as "used" as is discussed more fully below in relation to step]

708). If the date stored in the Insert Date field 57 does not ” fall within any of the "unused" "retention working list" date

ranges, then program control continues with step 712 Where the Backup Instance Record 42 is deleted. Otherwise, program

contrél continues with step 708 where all "retention working

list";entries satisfied by the date stored in the Insert Date

Field257 are marked as "used" to indicate that a Backup ‘ I Instance Record 42 has been used to satisfy this entry. This ensures that an older Backup Instance Record 42 is not used to satisfy a retention pattern specification when a newer entry also satisfies the condition. The DiStributed Storage Manager program 24 also checks to ensure that the file associated with the Backup Instance Record 42 has not been deleted prior to the "end date" of the window satisfied by the date stored in Insert Date field 57.‘ This condition is satisfied by ensuring that

the date stored in the Delete Date field 56 of the Backup Instance Record 42 currently being processed is after the "end date" of the window satisfied by the date stored in Insert Date

field 57. If the file was deleted prior to the "end date" of

the window, then the file cannot be used to satisfy the “retention working list“ entry since that file did not exist on ,the "end date". Following either step 708 or step 712, program control continues with step 710 where the Distributed Storage
‘ Manager program 24 determines whether there are any additional

10

Backupllnstance Records 42 associated with the File Identitication Record 34 currently being processed. If~so,.

program control is returned to step 704; otherwise, program control is returned to step 700.
While the present invention has been described in [connection with an exemplary embodiment thereof, it will be understood that many modifications and variations will be readily apparent to those of ordinary skill in the art. This disclosure and the following claims are intended to cover all such modifications and variations.

We claim:'

E, £312:’::>_ 1. A system for distribu d management of the

storage space and data on a network d computer system wherein
the networked computer system incl des at least two storage
devices for storing data files co rised of one or more binary

objects, said distributed storage management system comprising:
means for selectively c pying the binary objects

stored on one of the storage dev'ces to another of the storage

devices;
means for calculating a current value for_a binary
object identifier for selected inary objects stored on the storage devices wherein said c 0' lculation of said binary object identifier is based upon the, tual data contents of the associated binary object;
means for storing aid current value of said binary
object identifier as a prevf us value of said binary object identifier;
\

means for compar g said current value of said binary

object identifier associat d with a particular binary object to one or more previous valu s of said binary object identifier

associated with that particular binary‘object; and
means for comm nding said means for selectively
copying binary objects n response to said means for comparingg‘l

2. The distributed storage management system of
claim 1 wherein said means for calculating said our t value for said binary object identifier includes m

calculating a current value for a bina object identifier comprised of at least two ind ntly determined values;

3. The di ibuted storage management system of
claim 1 where“ said means for calculating said current value

for said binary object identifier includes means for calculating a 128—bit binary value comprised of four 32-bit

fields.

4. The distributed storage management system of

claim 3 wherein said four 32-bit fields include a binary bjeot identifier size field, a Cyclical Redundancy Check numb r field

calculated against the contents of the binary object ssociated
with said current value of said binary object ident'fier, a

Longitudinal Redundancy Check number field calcul ted against the contents of the binary object associated wi h said current value of said binary object identifier; and a binary object hash number field calculated against the co tents of the binary object associated with said current value of said binary object

identifier.

5. The distributed stor e'management system of
claim 1 wherein the networked c uter system further includes at least two local computers which is in communication

distributed storage manage nt system is cooperatively
executable on at least t of the local computers.

6. The dis ributed storage management system of
claim 1 wherein th networked computer system further includes a remote backup f le server and wherein said means for selectively cop ing binary objects selectively copies binary objects store on one of the storage devices to another of the

storage dev'ces and to the remote backup file server unless
sufficie storage space is not available on another of the

storage devices in which case the binary objects are copied

only 0 the remote backup file server.

‘6?

7. The distributed storage management system of
claim 1 further including means for auditing the perfo nce of
said distributed storage management system, said mea for auditing including: .

cowl/«g
second means forkeemmanéing said mean for
selectively copying binary objects to recopy previously
copied binary object; ‘ , \
means for recalculating said b ary object identifier

for the recopied binary object;
means for comparing said r calculated binary object
identifier tO'a previdus value of. aid binary object identifier for the recopied binary object; nd
means for reporting a failure if said recalculated
binary object identifier ' not identical to said previous
value of said binary ob' t identifier.

8. The di tributed storage management system of

claim 1 wherein sa'd means forfiééggggéggéflggid means for selectively copy'ng includes means fori§§§§3g§i£§%§aia means
for selectivel copying to copy a particular binary object only
if said curr nt value of said binary object identifier for that -

particular binary object is not identical to a previous value

of said inary object identifier for that particular binary

objec in response to said means for comparing.

9. A system for distribute anagement of the

storage space and data on a ne rked computer system wherein.
the networked computer s em includes at least two storage deviCes for storing ta files comprised of one or more binary objects, said ’ tributed storage management system comprising: ans for segmenting the binary objects into granules

7

‘ means for selectively gopying said granules stored on
one of the storage devices to an ther of the storage devices; g means for calculating a current value for a granule identifier for selected granul s stored on the storage devices “wherein said calculation of sa'd granule identifier is based upon the actual data contents of said associated granule;
means for storing aid current value of said granule
identifier as a previous va e of said granule identifier; means for compar‘ng said current value of said
granule identifier associ ted with a particular granule to one
or more previous values f said granule identifier associated

with said particular gra ule; and h

means for com anding said means for selectively
copying granules in res onse to said means for Comparing.

10. The distributed storage management system f‘
claim 9 further including means for reconstructing a inary
object from a most recent complete copy of the bi ry object, said means for reconstructing including:
means for copying said granules c ied by said means
for selectively copying granules to said ost recent complete copy of the binary object in order fr most-recently copied granule to least-recently copied g nule; and

means for generati ‘ bitmap for controlling said
means for copying said copi

11. The dis ibuted storage management system of
claim 10 wherein sa'd means for calculating said current value for said granule dentifier includes means for calculating a
32~bit Cyclic Redundancy Check number calculated against the

said g nule identifier and means for calculating a 32~bit

binary object hash number calculated against the contents
said granule associated with said present value of said” ranule

identifier.

12. The distributed storage management ystem of

claim 10 further including means for restoring a current version of a binary object to a previous versi of that binary object, said means for restoring including:
means for calculating said granul.identifier for
each of said granules in the current vers n‘of the binary object;
means for comparing said cal-ulated granule
identifier to a previous value of sa'd granule identifier for

each bf said granules in the curre version of the binary

object; and

means, responsive to aid means for comparing said
granule identifiers, for repl ing those granules in the current version of the binar object for which said granule identifiers are not iden

13. A syste for distributed management of the

storage space and da. on a networked computer system wherein the networked comp er system includes at least two storage devices for stori 9 data files comprised of one or more binary

objects and at east two local computers each of which is in

communication with at least One of the storage devices, said

distributed storage management system comprising:

eans for selectively copying binary objects stored

on one o the storage devices to another of the storage

means for indicating which of said copied binary
obj cts must be copied to a particular storage device in the

event of a failure of that storage device before that storag
device is considered to be operable by the local computer ith
which that storage device is in communication: and

c , , means fork, ‘ séid means for selecti ely
copying binary objects in response to said means fo indicating.

l4e The distributed storage manag ent system of
claim 13 wherein said means for indicating ncludes means for
specifying a last access date such that o ly binary objects 1
that have been accessed by the networke computer system on or
after said last access date must be c pied to a particular storage device before that storage.vice is considered to be operable.

,15; A system for di »ributed management of the
storage spaCe and data on a tworked computer system wherein the networked computer sy m in ludes at least two storage
. devices for storing data comprised of one or more binary objects, said distribut d storage management system comprising: means for s lectively copying the binary objects
stored on one of th storage devices to another of the storage
devices;
means or maintaining a file retention list wherein
said file rete ion list includes a file retention pattern for

each binary o ject copied by said meanshfor selectively copying binary obje ts; ’ eans for determining which of the binary objects:
copied b said means for selectively copying binary objects
match e ch of said file retention patterns; and
means for deleting the binary objects from the
stor ge devices in response to said means for determining.

. a“? <

‘ identifie

16. The distributed storage management system 0
claim 15 wherein said file retention pattern includes ily,
weekly, monthly, quarterly and yearly retention pa erns.

17. A system for distributed man ement of the
storage space and data on a networked co puter system wherein the networked computer system includ a remote backup file

server and at least two storage vices for storing data files comprised of one or more bina objects, said diatributed storage management syste mprising:

means for sel‘ tiv y cepying the binary objects

stored on one of the storage devices to another cf the storage devices or to the remote backup file server; and
mean for employing user-defined priorities to

determine w ch binary objects are to be copied to another
storage vice and to determine a queuing sequence for copying
binar objects to the remote backup file server.

///>” d 18., A method for mana ment of the storage space and

ata on a computer system wh ein the computer system includes
at least two storage area for storingfdata,files comprised of one or more binary obj cts, said method comprising the steps

of:

select ely copying the binary objects stored in one
of the storage areas to another of the storage areas;
oulating a current value for a binary object
for selected binary objects stored in the storage

areas erein said calculation of said binary object identifier
is b sed upon the actual data contents of the associated binary

ob'ect:

storing said current valu of said binary object
identifier as a previous value c said binary object identifier;
comparing said.c rent value of said binary object

identifier associated wi a particular binary object to one or

more previous values of said binary object identifier , associated with that articular binary object; and

controlli said step for selectively copying binary
objects in response to said step for comparing.

The method of claim 18 wherein said step for
calculating s 'd current value for said binary object identifier inclu es the step for calculating a current value
for a binary objec identifier comprised of at least two independently determi ed values.

20. The method\<: claim 18 wherein said step for calculating said current va ue f r said binary object
identifier includes the st or ca-culating a current value

i

for a binary object ident fie tili ing a 128~bit binary value

comprised of four 32~bit fiel sfléaa wherein said four 32~bit
fields include a binary object 1 n ifier size field, a

CycliCal Redundancy Chec mber fie d calculated against the contents of the binary object associat d with said binary
object identifier, a Longitudinal Redund ncy Check number field calculated against the contents of the bin ry object associated with said binary object identifier, and a‘bi ary object hash number field calculated against the contents f the binary

object associated with said binary object ident'fier;

Avie ‘

5 computer system includes at least two storage devices for

MW

\

wP/L' ’ ' 0! OLU/I‘ICLV
y. Idistributed storage management system further includesAammuKr

15 for storing the current value of the binary object identifier‘
in ‘ amoWr Ail/{W

COVQL
4 means

“Ade/ism
w

method for the distributed management of the storage space and
data on a networked computer system wherein the networked

bstoring data files comprised of one or more binary objects.

\yvz\ The distributed storage management system includesAneeae for
K selectively copying the binary objects stored on one of the

fi&&;t0rage devices to another of the storage devices andAmeeae for
lo Avxcalculating a current value for a binary object identifier £05.;
selected binary objects stored on the storage devices wherein
the calculation of the binary object identifier is based upon
the actual data contents of the associated binary object. The

Vfifiofias a previous value of the binary object identifiernflnuanwa for {Kflétomparing the current value of the binary object identifier

associated with a particular binary object to one or more
previous values of the binary object identifier associated with

zoggfifi that particular binary object andArnafis for commanding the

VV/ V v "Ll/x] L. \L/
answggcg OF TKE‘QISCLOBURfi.

The present invention is directed to a system and

0‘ ciwme‘

anoihf gin/{ct

“Au/ta.

for selectively copying binary objects in response to the for comparing.

Snmou

20¢ mom LecAL.

/‘‘5 my

I , l5\: COM
DSK J

‘3 DssK «”9

WIDE

\o A” . AREA NWOQK

(Sx

(““pkfissan awe/«3E

F‘Qez DISK smug

v v uuv/

VU V j

EECOMTVPE .. '- 3C,

EECoQO Fuse MAW: ‘. 34 -r4'
M ANAL-:EHENT Cums _" 43

4+

. 8mm? aqua IDENMQJK ,. «~44, _ amp,m -w*8
1N3m~c€ LAST Mommas DATE, firms. . V“ 50
QELQQD .
4?. mar Access DATE fume #52.

FM. xmmmras J* 5*
4‘58

, UNKT§ BACKUP mmme Ramp *" ‘50

8mm ,, r .,., ‘ * ‘ amid ”x. BINARY 63mm STQ‘EAMWOE. _~r~ 62.

Ileficmon ,- ' '
EEG-339.0 ' BINAKYOBTECT SQE. ”mm-

55 Swami OSTEQT cage);. ' «- Cato
8”qu IBINARV 063w L‘RC, v G8

0536‘: Swan carted HASH ‘ ‘ ‘ ‘.w 70 EWIQEL ‘ ' ‘
7+9 -m

ifRein

25

BACKUP

QQSUE
Qecoea

Pike A‘Trmmmas

. mus. enemy '

SCAN

[)5 L

f(02..

jam CHE— -
'Dem‘F‘CA‘th
9:va

lDE-JJV lF'k CRHOQ
KELOQD

meme ../ i30
Bacon '

:wANcfiemp

f I52. / I34 ('38)
cue/m: Bwam
06m:

(Mm IF'ICA'T'on
Racows '

.I4C

SEX:N QN‘T DAT A IDENTIFY

STQQAH INTO BIN/MEL lac"? NUCNPLE. 033'ng F62. ARV. 5'“ ~ Beam? 0636.613

‘ -~ J

2.00) . - . Con PtLe US‘T 0F
Batu/“LY CHESS

TO BE BACKS!)

242/
DEUSTYE 6‘va
QEflECT FROM
(.ng
5mg.

FIG. 58
. ' t

DON'T" Foil
MESSAGE F‘QQH
flestd;
ALLOCA'nw

‘BtNARV CKTQG'S

Fisk . BACKUP

FIG. SD

32%:

Sfiwb 4
HW’TO
Emacs
AkKOCA'fiN
KOUTIN E.

FIGSE

330)

Paxoamzmm
Romme ‘

(9102633wh3
wwesr 2mmNE. A5 Pmom‘f
BI MOW

0833.1“ AVNLMBUE.

'To CDHPLE‘SSxN

WINE. ‘

@uafieii BNA‘W
OBJECT bSEbHENT
OF D‘fiABtfléE HLE.

Umem.
@mam

IDEN’TWWEQS

41¢»)

wmm

Swmw
Fur:

FIGSG

418)

wrong

CQEME
Brr HAP

GM vmmmb

CDPVNOF: En we.
.QB'JECT

' 430
/

63mm Us? a?
GQAwwkfis

- . wvbbmw

eta—raw 442

mwnv 0“” ,/
F51 LI:

goHWLE. km q: q 4‘ 3

Bwa 0556133
N 9%moas ./
VERMN Ci. PVLE

gamma
(QMENYS

I DEN?! F”: 6.95

~ TWSHH‘
Gameca‘CD
Loam.
WEE.

RANDOML‘I

Sims Bump] ‘
'5'... LT

/502

CALLQWE:
E; NADA!
OBZTQCI

(0&9? ‘ EEK

5&0

‘ /
Gawm
AUDIT aware
EMENT

F16. ST

goo OGTAN LAST
\. gAmsss Dams.

‘ 9655mm:

907— Fufi. Mums;

N. R69“ D

5,04 F'xLE.
’ um‘cmmog

Pm om: DQMS.
ON LNG

mm. H051“

"' Rags»? 8mm)?
(9‘0 mm mugs;

um’TMSTG, harm

95' (11 be.) SET
mean-non STATUS
‘Vo 't NoQNAL,"

6": ‘56:? HIbQA'Wow
*xm‘ms To

“WW‘TQD”

HG. 5K

”IOD\

Locme. EM
Ft LE.
KOEN'MFKXTMN
Rm

Regen snow?
I N STANCE
Recon

EECOM'WPE. 3-9.

Fins, A ' CILE Locmum ‘ __ 38

'Ibemxrnaarow fl _ J, 40
RECORD F'le. NAM:
34 W Game-q 31mm) 3‘ 4!

H AIQAGENENT cums f‘ 43

|* a

LNL’TO FxLE inemmgmw macaw 4dr

gnaw? Cw (n.5,:bemmee‘ «‘44:,

BACKUP /“~. FlLE SVZE . ' “'" ”f8
1N5“WC?— um Momma!) DATE frmE "‘ 50
{EQQQD ‘ 1 '
41 LAST Access DATE ME MSZ

Fame. Ammames ‘ $54

Damn DATE. ‘ f" '54;

stem DATE. 3‘57

. Umbra 8mm? INSTAMCE R6032!) ”" GO
BINAK"! . . -

omacv m BlmAfié/ 033m STQQAMWDS 3/“ 62.
IDEXITIHCA‘HOQ - ' ' ” ,. s

EEC-DRY) BfiNAL‘f 083.36.? 3‘15 'v :"'€4‘

58 BIN/Mu owed CRCSZ. _v- (4,

' 053* T ‘ v

61mm VB‘NARV 7,, EC— LJLC. iiiiiiiiiiiiiiiii fi (38
(>de Emma; amecw HASH . 3’" “to

TIMWEL
7+ ‘SN’WN QKFfiCT arms“? w ”22.

HQ?)

25

[LE LOCK‘HOKX

F1 LE NAME.

BACKUP Page, “Smus

QUEUE 1.” ‘ RECORD ' F‘ L33: 'Sl'lt..
75

93$

.rvv 5%;wa Nuxzux;

DAT AGASE

F‘G SA

v - uu\

(teams. PMS.
mam“ u PICA’ON
Recon

‘DFJJT lFt CNIOQ
RELOQD

QR€A1€ ../ I30
8&{09
INSTANQE.m

f as; 98)
apname FMS.
'IN‘TO DATA
S‘TQEAMS

(meme; B: mm
Ofirezj
mun urn/{Paw
Rficokbs‘ '

.040

SQBNQN‘T MTA
*L “sTQQA tN—ro

[36° NWTIRE
EBWAQV
OVXTF6 ”(Q

2.00)
CoHPcLE US"? OF

Bum/XE)! OBTEELJS
To BE Emcm'ib‘

, BICKOP

'TQHNSWT -
3mm“. PNE—
are REHSTE *

8mm)?

/ 30E:

SEND. MESSAGE
To Rescued;
ALLOCATION)
ROUTINE

w/MT‘ Fog
HESS AGE WON
265m620?.
Auger. Know

ammo, mags
(TOR

2,le
Degas. 64mm!
(DESEG mam
LOCAL

STORM-,3

FIGSD

Seam!)
HESSAGG. 1’0

RESOJQCE
AkKOCJx’fiW
ROUTINE.

FIGSE

WNT F‘OK
Nessm FROM
mama/mm

CQHOQESSwQ'
Roan we Ab
AVA \LABLE

H‘GHEZS‘V Pmoerv‘f
BINAQV

AVA\\.§%\.E

1 HESSAGE TD

MKQP /Res109£
ROUTINE.

N - ‘ ' 3Q.

360)
my Sent) Message.
usuaEL, Y ”YD DELEJ‘E

PRuoRfl‘! ‘ Lama. o
F15... F7 LES

{DemlFY
WRGNEK RNA”!

CWTVQJVS
! 05.th $538315

was

lDEN’TxFIEQS ‘

meag—

430

came Lm OF
GRANULES‘

CQEA'VE
41+

L, arr HAP

4740
k we HOST
RECENT

GRANULAUZEJ) (DEN OF“ B: HARM

HGSH

OBTAN 442
wanm! 0" J
FILE;

<9ng MS? or:
BINARY ONWS E *3
N Fbtmms /
VERNON) CF Flue

QAKLUWE 444
CONTENTS /

IDQN'TI'FVEQS

' TRANS Hfl'
GQRvaETO

F‘lQSI

{N Ume
Rama 09’
RmmHL‘f
52mm» 8mm»!
OKkQ T

CMCUWE

E51 NADA]
0836121"
£06351 I Rea

SK)

GEWTE;
AUG IT FAKMIE
EMT

H353“

@v5 3 SJ 3“ng

goo OQTAN um?
\. Access DATE:

N Rama

607— Qua DATABAS‘E

4290
F1 L6,,
[OEEXSTFlCAvaJ

(904

PUT Dun DQNE‘.
ON LAME

W16. HDST
"" Regem 8mm?
élo N31 «\qu

\N‘TI‘STE. Q€STOQAT¥

C? FKLEjfim‘

msm‘nw SWWS ‘ “To "NORMAL,“

6% SET HibeA‘Ilok)
“MUS "To
“Hm-$213166 '

' FiGZSK

LDC—ATE. mm
Fuse.

mawmfiunMQ
Rflufib

“$2,;

”704)

‘5‘??? c N m:

fif§§i63 ”(UP gjggl 315“,

Rama)
USED.

newts.
Mama
\NSTANCE

Reamer:

FIGSL

‘ James R. Woodhill et al. : MANAGEMENT ON NETWORKED “At:
3 COMPUTER SYSTE%§WH R A
Serial No. 08/085,596 : ginEmy %?N N‘
‘ “Vt {ti 3}?

. z ,' LNG . i“ Filed July 1, 1993 Gronp No 23oohwy 75 $95

3: $3M? in} Wm; 21’ f? “'6 fifiifgd

AMENDMENT

PATENT
93085

Art Unit 2307 L Z61.
Examiner Homers ; :2; /<E;#
In re application of : SYSTEM AND METHOD F R
DISTRIBUTED STORAGE

IN THE UNITED STATES PATENT AND TRADEMARK OFFI

Pittsburgh, Pennsylvania 15222
February 20, 1995
Hon. Commissioner of Patents
and Trademarks

Washington, DC 20231
Sir:

In response to the Office Action dated 21 October 1994,>~j please amend the abovesidentified application as folloWs:)

In thg title

Please amend the title of the above—identified application
by adding at the end thereof ¥~Using Binary Object Identifiers~—.

In the claims.
Please amend the Claims as follows:

1. (Once Amended) A system for distributed u “agement of

the networked computer s« ‘des at least two storage
devices for storing data omprised of one or more binary
Objects, said dis . age management system comprising:
means or selectively copying the binary objects stored on

one 9,' he storage devices to another of the storage devices;

Cad

memwwmu~amm

‘ selectively copying binary objects in response to said means for

means for calculating a current value for a binary ob'-ct identifier for selected binary objects stored on the arorage devices wherein said calculation of said binary oa'ect identifier is based upon the actual data contents of the.ssociated binary
object[;l s.id c.lcu at-d binar ob'ect fienti ier.ein- u ed as

the ioentifi~r for th— s-lec ed bin.r ob'ect-

means for storing said..e value of said binary object
identifier as a previous valu: f said binary object identifier;i

means for comparing s-Jd curr: t value of said binary object identifier associated wi»h a oartfoular binary object to one or more previous values of S~ d binary object identifier associated

with that partic var binary object; and
means fo [commanding] controlling said means for

comparivg.
i

Claim 7, line S, cancel "commanding" and substitute therefon ~«contr6lling~~.

Claim 8, line 2, cancel "commanding" and substitute therefoJ ——contfolling~—.

Claim 8, line 3, cancel ”commanding" and substitute therefod -~inst;;;tingwe.

the networked computer system inclu least two storage devices for storing da es comprised of one or more binary
objects, sa' istributed storage management system comprising:
same for segmenting the binary objects into granules of

data;

.previous val *s of said granule identifier associated with said

/'” "

means for selectively copying said granules stored on/e, of

the storage devices to another of the storage devices;
means for calculating a current value for a gr. ule
identifier for selected granules stored on the storage devices wherein said calculation of said granule '6entifier is based upon the actual data contents of said asso’iated granule[;]. said

c.lculated oranule identifi- b*' OHWSBdWaT the identifier for“ WW

means for Storing s='d current value of said granule
identifier as a previous v. ue of said granule identifier;
means for comp.ring said current value of said granule identifier asso ated with a particular granule to one or more

particular granule; and
me. 8 for [commanding] contrglling said means for
seler ively copying granules in response to said means for
c. paring.

Claim 13, line 16, cancel "commanding" and insert therefor ~~controlling——.
«mm—”W i-
/

18. (Amended) A method for management 0 the storage space
and data on a computer system wherein the omputer system includes at least two storage areas r storing data-files

comprised of one or more
the steps of:
selectively copyin / e bi ary objects stored in one of the storage areas to an her

e identifier

storing said current value of sai inary object identifier
as a previous value of said object identifier;
comparing said curr nt us of said binary object
identifier associated wi 'cular binary object to one or
more previous values ary object identifier associated
with that partic object; and
. control ng said Step for selectively copying binary objects
in respo e to said step for cemparing.

Remarks

In response to paragraph 1 of the Office Action, the title
of the invention has been amended to indicate that binary object

identifiers are used in the management of storage space on.a network computer system. It is applicants’ position that the

title, as amended, is descriptive of the invention. If the
Examiner remains of the position that the amended title is not indicative of the invention, the Examiner is respectfully requested to suggeSt wording changes to the amended title.
In response to paragraph 2 of the Office Action, applicants acknowledge that the instant application has been filed with informal drawings. Formal drawings will be submitted when allowable subject matter is indicated.

In paragraph 3 of the Office Action, claims 1, 9, and 13
stand rejected under 35 U.S.C. § l12 because the phrase "means for commanding... comparingfi is alleged to be ambiguous.

Claims 1, 9, and 13 haVe been amended to recite "means for
controlling". The means for controlling controls the means for selectiVely copying in response to the means for comparing. As

amended, it is believed that the 35 U.S.C. § 112 rejection is new
overcome;

PB/WEMWO

PATENT APPLICATION SERIAL NO.

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE
FEE RECORD SHEET

070 DF 12/18/95 0855537é - . 1 101 750.00 CK

Attorney’s Docket No.W ' PATENT

in THE UNITED antes enam AND manemanx oerxcaé): l.o? (/
Anticipated Clasaification of this application: ClassWSubclass

Application No.: 0 8/who“.

PRIOR APPLICATiON Examiner:we, ‘7'

I Art Unit:

W

Box FWC
Assistant Commissioner for Patents

Washington, {10.20231

FILE WRAPPER CONTINUING APPLICATION (FWC) TRANSMITTAL’
(37 C.F.R. 1.62)

WARNING: This form cannot be used where the parent case may not be abandoned because the filing of e
' request under the FWC procedure "will be considered to be a request to expressly abandon the
pn‘or application as of the tiling date‘grenteo’ to the continui application." 37 OF‘R, 162(9)

WARNING: This procedure can only be used for a pending application prior to ayment of the issue fee (37
GER. 762(3)). exceptit the parentappiicetion was withdrawn under 37..R. 7.373(b)(5) “to permit
consideration of an information disclosure statement under 1.9? in a con ' uing application. " 37
C.F.F~i. 1,62ta). '

WARNiNG: The filing of an application at the United States stage of an international appiic 'on requires an
oath or declaration 37 OFF, l.61(a)(4).

WARNING: The claims of this new application may be finally rejected in the first Office action where all claims
of the new application are drawn to the same invention claimed in the earlier application and would
have been properly finally rejected on the groands or art of record in the next Office action if they
had been entered in the earlier application. MPEP § 706.07(b).

WARNING: An application under 37 CPR. 1.62 is filed by making changes by amendment to the prior
app/ioaiion,'(37 ORR. l.62(a)), and not by tiling a new application, 37 ORR. i.62(e). ..

WARNING: Filing under 37 C,F.R. 7.52 is pennitted only if filed by the same or less than‘ail the inventors named
in the prior application. 37 CFR 1.652(3). '

cenrtrlcanou luuoeao'r C.F.R. 1.10

i hereby certify that this FWC Transmittal and the documents referred to as attached therein are being deposited with the United States Postal Service on this date November 9 1 9 9 5. in an envelope as “Express
Mail Post Office to Addressee." mailing Label NumberWdemssed to the: Assistant Commissioner for Patents. Washington, 0.0 20231.
‘Beth H. Retort

. arson mailing paper)

/~

(
NOTE: Each paper or fee filed by “Express Mail”7771?? have the number of the “Express Mail" mailing label placed thereon pn’or to mailing. (37 C.F.Ft. 1.1 C(13)).

WARNING: Certificate of mailing (first class) or facsimile transmission procedures of 37 C.F.R. 1,6 cannot be used to obtain a date of mailing or transmission for this correspondence.

. Sit'~ature of parse mailing paper

(FWC [4n2]'-—page 1 of 13)

WARNING: File wrapper continuing procedure filings can only be based on a prior complete application as
defined by § l.51(a)(il, and not a prior complete provisional application as defined by § 1.5 1(e)(2,l.

This is a request for e filing under the file wrapper contlnuing application procedure (37 C.F.R. 1.82), for a .
C3 continuation
D divisional

C} continuatl0n~ln-part (for oath or declaration, see Ill below)
Attached is an amendment for added subject matter
[3 continutng applicatton to permit consideration of an information disclosure
statement under 37 C.F.R. 1.97.

NOTE: The filing date under 37 Cffl. 1.62(a) is ”... the date on which a request is filed {or an application
. . including identification of the application number and applicant’s name of the prior application."
The prior application under 37 C.F.R. 7.652(8) must be “... a pn'or complete application," as defined
in 37 C,F.R. 1,51(a)(1).

PARTICULARS OF PRIOR NONPROVISIONAL APPLICATION

WARNING: File wrapper continuing procedure filings can only be based on a prior complete application as
defined by § 1.5 Halli), and note prior complete provisional application as defined by § 1.51(a)(2).
37 C.F,R. 1.6263),

A. Application No. 08 / 085,596 ‘. filed Wgéate.
B. Title (as originally filed SYSTEM & METHOD FOR DISTRIBUTED STORAGE
and as last amended)....
C. Name of applicant(s) (a
correspondence addressof applicant(s)

(FWC [441—49398 2 of 13)

W0

W‘NEGEMENI 0N NE I WORKED ' COMPU I‘ER S SE STEMS

, "WQEW

I. FULL FAMILY NAME FIRST GIVEN NAME
NAME OF

xwwwmx Woodhill James

RESIDENCE & CITY STATE 0R FOREIGN COUNTRY OF CITIZENSHIP
CITIZENSHIP COUNTRY .

Houston Texdgw7rmi U.S.A.
POST OFFICE STATE & ZIP CODE/COUNTRY ADDRESS

TX 77019

POST OFFICE ADDRESS CITY

1960 Haddon
Avenue Houston

2. FULL FAMILY NAME FIRST GIVEN NAME SECOND GIVEN NAME
NAME OF

INVENTOR

git) Louis 'W. W

WOodhill

RESIDENCE &
CITIZENSHIP

STATE 0R FOREIGN
COUNTRY

COUNTRY OF CITIZENSHIP

MRichmond Texas. U.S.Ag

POST OFFICE
ADDRESS

POST OFFICE ADDRESS

2114 Pecan
Trail Drive

STATE & ZIP CODE/COUNTRY

Richmond Texaé 77469

3. FULL FAMILY NAME FIRST GIVEN NAME SECOND GIVEN NAME
NAME OF
INVENTOR

W

More, Jr.
W

William

W‘IRuSsell

flureonyoxch
COUNTRY

RESIDENCE &
CITIZENSHIP

COUNTRY OF CITIZENSHIP

~.. WWHouston Texas

POST‘OFFICE

ADDRESS‘ ‘

\STATE & ZIP CODE/COUNTRY POST OFFICE ADDRESS

786 Thicket Texas 77079

[g] continued on Added Page for Inventor’s Data

wwcmaymmeam1m

_\\

FAMILY NAME FIRST GIVEN NAME SECOND GIVEN NAME

Berlin Harris“.- w...»

RESIDENCE 8L
CITIZENSHIP

STATE 0R FOREIGN
COUNTRY

COUNTRY OF CITIZENSHIP

POST OFFICE
ADDRESS

5. FULL NAME
OF INVENTOR

RESIDENCE & CITY
, CITIZENSHI?

POST OFFICE ‘POST OFFICE ADDRESS
ADDRESS

64 FULL NAME
OF INVENTOR

RESIDENCE & CITY
CITIZENSHIP

POST OFFICE POST OFFICE ADDRESS
ADDRESS

POST OFFICE ADDRESS

9111 Landdown-
Drive

"TATE & ZIP CODE/COINTRY

TX 77096

FAMILY NAME FIRST GWEN NAME SECOND GIVEN NAME

STATE 0R FOREIGN
cmmmY

COUNTRY OF CITIZENSHIP

STATE & ZIP CODE/COUNTRY

‘ FAWLY NAME FIRST GIVEN NAME SECOND GIVEN NAME

STATE on FOREIGN COUNTRY OF cmzawsmp
cmmm?

STATE & HP CODE/COUNTRY

(FWC I[4-2.1]-—-Added Page)

The above identified application, in which no payment of issue tee, abandonment of (other than where the above identified application was abandoned under 87 C.F.R. 1.318(b)(5) to permit consideration of an information disclosure statement under 37 C.F.R. 1.97), or termination of proceedings has occurred. is hereby expressly abandoned as of the filing date of this new application. Please use all the contents of the prior application file wrapper, including the drawings, as the basic papers'tor the new application.

it is understood that secrecy under 35 U.S.C 122 is hereby waived to the extent that if information or access is available to any one at theapplications in the tile wrapper of a 37 C.F.R. 1 L62 application, be it either this application or a prior application in the same file wrapper, the PTO may provide similar information or access to all the other applications in the same file wrapper.

II. lnventorshi'p statement

NOTE: “it the continuation. r'continuationjln-part, or divisional application is filed by less than all the inventors named in the prior application a statement must accompany the application when filed requesting
deletion oi the names of the person or persons who are not Inventors of the invention being claimed
in the continuation, continuationwinnpart, or divisional application. " 37 CiFfl. 1.62(a).

NOTE: "in the case at a continuation-impart application which adds and claims additional disclosure by ‘ amendment, an oath or declaration as required by § 1.63 must be filed. In those s/tualions where a
new oath or declaration is required due to additional subject matter being claimed, additional inventors
may be named in the continuing application. in a continuation or divisional application which discloses
and claims only subject matter disclosed in a prior application, no additional oath or declaration is required and the application must name as inventors the same or less than all the inventors in the prior application." 37 OER 7.60(C). ‘

{complete applicable item (a), (b) and/or {0) below)

(a) (3 This application discloses and claims only subject matter disclosed in the prior
application whose particulars are set out above and the inventorls) in this application are
8: the same.

C less than those named in the prior application. it is requested that the
following inventor(s) identified above for the prior application be deleted:

(type namels) of inventor(s) to be deleted)

(b) [3 This application discloses and claims additional disclosure by amendment and
a new declaration or oath is being filed. With respect to the prior application
whose particulars are set out above, the inve‘ntor(s) in this application are
Ci the same.

[1 Add the following additional inventorls).

(type name of inventoris) ‘to be added)_

(c) The inventorship for allvthe claims in this application is

El the same.

i] not the same. An explanation, including the ownership of the various claims
at the time the last claimed invention was made, is submitted. '

(FWC [4-21—page 4 of 13)

lil. Declaration or oath

A. Continuation or divisional

33 None required.
8. Continuationvin~part
[3 Attached.

Executed by

(check all applicable items)

D inventor(s).

{:1 legal representative of inventorls). 37 C.F.R. 1.42 or 1.43.

[3 joint inventor or person showing a proprietary interest for inventor who
refused to sign or cannot be reached. 37 C.F.R. 1.47;

D This is the petition required by 37 C.F.R. 1.47 and the statement required by 37 C.F.R. 1.47 is also attached.
. (See item Vlll below for fee.)
[3 Not attached.

{:1 Application is made by a person authorized under 87 C.F.R. 1.41 (c) on behalf of all of the above named 'applicant(s).
(The declaration or‘oath, along with the surcharge required by 87
C. F. Fl. 1.1603) can be filed subsequently.)

(:3 ”Attached is a showing that the filing is authorized.
(Not required unless called into question. 37 C.F.R. l.41(d))

W., identification of Ciaims for Further Prosecution

WARNING: "The claims of a new application may be finally rejected in the first Office action in those situations where (1) the new application is a continuing application of, or a substitute for, an earlier application,
and (2) all the claims of the new application (a) are drawn to the same invention claimed in the
earlier application, and (b) would have been properly finally rejected on the grounds of art cl record
in the next Office action if they had been entered in the earlier application." MPEP § 706.07lb).

@ The fees to be charged are to be based on the number of claims remaining as

a result of the:

6 attached preliminary amendment.

E] the unentered amendment filed under 37 C.F.R. 1.116 in the prior applica-
tion, which is now repeated.

L: the claims as on tile in the prior application.

" (ch [4-2]——-page 5 of 13)

V. Fee Calculation (37 C.F.R. 1.16)

NOTE: The filing fee for a continuation, continuation-in-part,- or divisional application is based on the number
of claims remaining in the application alter entry of any preliminary amendment and entry of any

amendments under 37 C. F. H 1.116 unenteredIn the prior application which is requested to be entered in this FWC application. 37 C. F. R. 1 62

CLAIMS FOR FEE CALCULATION

Number Filed Number Extra Rate Basic Fee

37 CRR. 1.16(a)
$750.00

Total Claims 4
(37 cam. l.16(c) 4.. 20.= 0 x, 3; 22.00

Independent Claims 0 V ‘
(37 cm. 1.160;» 2 —~ 3 = x $ 78.00

Multiple dependent claim(s). if any .
(37 C.F.R. 1.I6(d)) _ . + $250.00

E] The fee for extra claims is not being paid at this time..

Filing feecalculation himw
NOTE: If the fees for extra claims are not paid on filing they must be paid or the claims cancelled by amendment,
pn‘or to the expiration of the time period set for response by the Patent and Trademark Office in any
notice of fee deficiency. 37 C.F.R. t.16(d).

Vi. Petition for Suspension of Prosecution for the Time Necessary to
File an Amendment

NOTE: Where it is possible that the claims on file will give rise to a first action final for this continuation application
and for some reason an amendment cannot be filed promptly (e.g., experimental data is being gathered),
it may be desirable to file a petition {or suspension of prosecution for the time necessary.

(check the next item, if applicable)

[Ii There is provided herewith a Petition to Suspend Prosecution for the time
Necessary to File an Amendment (New Application Filed Concurrently).

VII. Small Entity Statement

[Z] A verified statement that this is a filing by a small entity is attached.
WARNING: "Status as a small entity in one application or patent does not effect any other application or patent, including applications or patents which are directly or indirectly dependant upon the application
or patent in which the status has been established A nonprovislonal application claiming benefit
under 35 USC 119(6). 120,121 or 365(0) of a prior application may rely on a verified statement
filed in the prior application if the nonprovisionel application includes a reference to a verified
statement in the prior application» or includes a copy of the verified statement tiled in the prior
application if status as a small entity is still proper and desired. " 37 C.F.Fi. § 728(3).

(ch [4-2]—-page 6 01 13)

(complete the following, if applicablei

E] Status as a small entity was claimed in prior application
WI , flied on , from which benefit is being-clalmed for this application under.
35 U.S.C. D 120.
E] 121,
[3 365(0),
and which status as a small entity is still proper and desired.

[2] A copy of the verified statement in the prior application is included.

Reduced filing fee calculation (50% or above) s

NOTE: 37 C.F.R. 1.28(a) states “Status as a small entity must be specifically established by a verified statement
filed In each application or patent in which the status is available and desired, except those applications
filed under § 1.60 or § 1.62 of this part where the status as a small entity has been established in a
parent application and is still proper. " '
The last sentence of 37 C.F.R. 1.28(a) states “Applications filed under § 1.80 or § 1.62 of this part
must include a reference to a verified statement in a parent application if status as a small entity is
still proper and desired.”
Any excess of the full fee paid will be refunded if a verified statement and a refund request are filed within 2 months of the date of timely payment of a full fee then the excess fee?paid will be refunded on request. 37 C. FUR 1..28(a)

VIII. Fee Payment Being Made at This Time Not attached

i3 NOlfiling fee is submitted

(This and the surcharge required by 37 C. F. R. t 16(e} can be paid subseqUently)

Attached .

5&1 filing fee . $15.9...92m
[3 recording assignment
($40.00; 37 C.F.R. t.2‘l(h)).
For payment of fee see item XIV beiow. $

[3 petition fee for filing by other than all the
inventors or person not the inventor where
inventor refused to sign or cannot be reached
1 ($130.00; 37 C.F.R. 1.47 and 1.1769) , $
Cl processing and retention fee ‘
($130.00; 37 C.F.R. 1.53(d) and 1.210)) $
NOTE: 37 C.F.R. 1.2m) establishes a fee (or processing and retaining any application that is abandoned for
' failing to complete the applicah'on pursuant to 37 C.F.R. 1.53m and this, as well as the changes to,
i 37 C.F.H. 1.53 and 1.78, indimte that in order to obtain the benefit 0/ a prior U.S. application, either
the basic filing fee must be timely paid or the processing and retention fee in 5 1.21m must be paid
within 1 year from the notification under § 1.58M). i

i Total tees enclosed $ 7 5 O ' 0 O

(FWC [4-21—page 7 of 18)

IX. Method of Payment of Fees

[X] Attached is check in the amount of 35 m

[:1 Charge Account No.Win the
amount oi‘$

Ci A dupiicate of this reouestiis attached.
NOTE: Fees should be itemiZed in such a manner that it is clear for which’pUrpose the fees are paid. 37 C.F.Fi.
1.22lb).

X. Authorization to Charge Additional Fees

WARNING: If no lee payment is made at this time, this item should not be completed.
WARNING: Accurately count claims, especially multiple dependent claims, to avoid unexpected high charges
if extra cialm charges are authorized.

[3 The Commissioner is hereby authorized to charge the foitowing additional fees

that may be required by this paper and during the entire pendenoy of this

application to Account No ”3.3.2.1319“:

£11 37 cFR 1.16(a),)(f) or (g) (filingvfees)
E] 37 C.F.Fi. 1.16(b), (c) and (d) (presentation of extra claims)-
NOTE: Because additional tees for excess or multiple dependent claims not paid on filing or on laterpresentation
must only be paid or these claims cancelled by amendment prior to the expiration of the time period
set for response by the PTO in any notice of fee deficiency (37 OER. 1.16(d)), it might be best not

to authorize the PTO to charge additional claim lees, except possibly when dealing with amendments after final action.

E 37 C.F.R. 1.16m) (surcharge for iiiing the basic iiiing fee and/or declaration
on a date later than the filing date of the application)
E] 37 C.F.Fi. 1.17 (application processing fees)
WARNING: While 37 C.F.R. 1.17(a), (b), (c) and (d) deal with extensions 01 time under § 1.136(5), this
authorization should be made only with the knowledge that. "Submission of the appropriate
extension lee under 37 C. FR. 1. 136(a) is to no avail unless a request or petition lor extension is
filed" (Emphasis added). Notice of November 5,1985 (1060 O. G. 27)

[j 37 C.F.R. 1.18 (issue fee at or before mailing of Notice of
Allowance, pursuant to 37 CPR. 1.31163»
NOTE: Where an authorization to charge the issue fee to a deposit account has been filed before the mailing

or a Notice of Allowance, the issue fee will be automatically charged to the deposit account at the time of mailing the Notice of Allowance. 37 CF. Fl. 1 317m)
37 C.F.R. 1.28(b) states: (a) notification of change of status must be made even if the fee is paid as
"other than a small entity" and (b) no notification is required if the change is to‘another small entity. Notification of any change of status resulting in loss of entitlement to small entity‘status must be filed
in the application prior to, or at the time of, paying the issue fee. 37 C.F.R. 1.28(b).

XI. Instructions as to Overpayment
[3 Credit Account No.
R} Refund

(ch [4-2]-page s of 13)

but. Priority-35 u.s.c. 119(a)-(d)

[:1 Priority of Application No. /...______..._ filed on
in. (country) is claimed under 35 1.1.8.0. 119.

D The certified copy has been filed on in prior U,S. applica
tion Serial No. 0 /W, which prior application was filed on

Ci Certified copy will follow.

XIII. RelateBack

WARNING: ii an application claims the benefit of the filing date of an earlier filed application under 35 U.S.C. 120, 121 or 365(c), the 20«year term of that application wil/Ibe based upon the filing date of the
earliest U. 8. application that the application makes reference to under 35 U.S.C. 120. 121 or 365(c).,
(35 U.S.C. 154(a)(2) does not take into account, lor the determination of the patent term, any
application on which priority is claimed under 35 U.S.C. 119, 365(8) or 365(b).) For a c-l—p
application, applicant should review whether any claim in the patent that will issue is supported
by an earlier application and, if not, the applicant should consider canceling the reference to the
earlier filed application The term of a patent is not based on a claimvby-claim approach. See Notice
0/ April 14 1995, 60 Fed. Reg 20,194, at 20, 205

NOTE: "Any nonprovisionai application claiming the benefit of one or mere prior filed copending nonprovisional applications or international applications designating the United States of America must contain or be amended to contain in the first sentence of the specification fol/Owing the title a reference to each such
prior application, identih/ing it by application number (consisting of the series code and serial number)
or international application number and international filing date and indicating the relationship of the

applications. Cross~reierences to other related applications may be made when appropriate. (See § 1.14031)" 37 CWFR § 1.78{2).

\ (complete the following, if applicable)
8] Amend the specification by inserting, before the first line. the sentence:
A.35 UWSC 119(6)

NOTE: “Any nonprovisicnai application cieirnlng the benefit of one or more prior tiled copending provisional applications must contain or be amended to contain in the first sentence of the specification following
the title a reference to each such prior provisional application, identih/ing it as a provisional application,
and including the provisional application number {consisting ofseries code and serial number}. " 37 CFF?
§ 1 78(a){4}

WARNING: While this application under‘37 C.F.F?, § 1.62 cannot be a file wrapper continuation application of a provisionai application, the nonprovisionai application giving rise to this FWC filing could claim
the benefit of a provisional application.

1:] “This application claims the benefit of Us. Provisional Applicationis) No(s).:

APPLICATION NO(S).: FILING DATE

/mm It

- (FWC [4-2lwpage 9 of 13)

e. 35 use. 120, 121 and 365(0)
NOTE: "Any nonprovislonal application claiming the benefit. of one or more prior tiled copendlng nonprovlsional
applications or International applications designating the United States of America must contain or be
amended to contain in the first sentence of the specification fol/owling the title a reference to each such
prior application. identifying it by application number (consisting of the series code and serial number) ’ I “ it
or international application number and international filing date and indicating the relationship of the i " '
applications Cross-references to other related applications may be made when appropriate (See
§ 1.14m." 37 can. § 1.743(2).
i3 “This application is a .

\ ')Ej continuation i

i i] divisional - ‘ A

C] continuationin-part

0 qiflf COpendlng application(s)
file] \X E] Serial Number 0 8/%filedWWQBQVIJOWA

J I" which designated the US"I , NOTE: The proper reference to a prior filed PCT application that entered the US. national phase is the US.

serial number and the filing date of the PCT application that designated the US.

DOV. Assignment

[E The prior application is assigned of record to
% Le ent Cor oration

D An assignment of the invention to

‘ is attached. A separate [2] “COVER SHEET FOR ASSIGNMENT (DOCUMENT)
l ACCOMPANYING NEW PATENT APPLICATION” or C] FORM PTO 1595 is also
attached.

NOTE: “ll an assignment is submitted with a new application, send two separate letters-«one forthe application
and one for the assignment" Notice of May 4, 1990 (1114 0.6. 7748).

XV. Power of Attorney

The power of attorney in the prior application is to

Christopher H. Gebhardt 33:130
Attorney Reg. No.
I , I.

a. X] The power appears in the original papers in the prior application.

b. E] The power does not appear in the original papers, but was tiled on
0. Cl A new power has been executed and is attached.
d. E] Address ail future communications to: ~

(item at may only be completed by applicant, or attorney or agent of record.) Edward L. Pencoske 29,688
Name A '. Reg, No.

Kirkpatgick 5: Logghart LLP , . Address

1500 Oliver Building (412) 355~8645
Tel No. -

Pittsburgh, PA 15222 I (FWC {4-2}~page 10 of 13)

RVI. Maintenance of Copendency of Prior Application

(this item must be completed and the necessary papers filed in the prior application,
if the period set in the prior application has run.)

if? A petition, tee and response has been filed to extend the term in the pending
prior application until”MW.

NOTE; The PTO finds it useful it a copy of the petition filed in the prior application extending the term for
response is filed with the papers constituting the filing of the Continuation application. Notice of Noven‘iber 5, 1985 (1060 0.6. 27).

K] A copy of the petition for extension of time in the prior application is

1 attached. ' ‘ XVII. Conditional Fetitions for Extension of Time in Prior Application
(complete this item and file conditional petition in prior application,
if previous item is not applicable)

[3 A conditional petition for extension of time is being filed in the pending prior
application
NOTE; The PTO finds it useful if a copy of the petition filed in the prior application extending the term for
response is filed with the papers constituting the filing of the continuation application. Notice of November 5, 1985 (1060 0.6. 27), ~

[:1 A copy of the conditional petition for extension of time in the prior application
is attached.

Wiii. Abandonment of Prior Application

X] Please abandon the prior application at a time while the prior application is
pending or when the petition for extension of time or to revive in that application
is granted and when this application is granted a filing date so as to make this application copending with said prior application, At the same time, please add
the words “now abandoned” to the amendment to the specification set forth
in Xlii above.

3 NOTE: According to the Notice of May 13, 7983 (103 W06 5- 7). the filing of a continuation or continuationain—
part application is a proper response with respect to a petition for extension of time or a petition to
revive and should include the express abandonment of the prior application conditioned upon the
granting of the petition and the granting of a filing date to the continuing application. l

1 NOTE: “A registered attorney or agent acting under the provisionsof § 1.34la}, or of record, may also expressly
abandon a prior application as of the filing date granted to a continuing application when tiling such
a continuing application." 37 C,F.R. i.138. -

XIX. information Disclosure Statement

[:1 Submitted herewith is an information Disclosure Statement.
x.

(FWC [4-2]——page ii of 13)

Assignment recorded in PTOon.QQEngr 21 g 1 993

6737 Fréme 0936

D PMS ADDED PAGE FOR INVENTOR’S DATAFOR FWC FiLING

D Pius ASSIGNMENT (DOCUMENT) COVER LETTER ACCOMPANY~
ING NEW PATENT APPUCATION

Reel

W.-

{ W“x ' SIGNATURE or amomey n x a. "

Reg. N 9? E) \,. ’K > /j 3.

QWW‘» «.33;ch-

‘ Edward L. Pencoske

(type or pn’nt name of attorney)

‘ Kirkpatric ,LLP

p,o.Address pittsburghr WpA 15222
WWW—W

W“

m...”

Tel. M11612) 355~8645

,7

(FWC [4-21wpage 13 of 13)

v.

”NU

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Art Unit
Examiner =

In re application of : SYSTEM AND METHOD FOR
- DISTRIBUTED MANAGEMENT ON
James R. Woodhill et a1. : NETWORKED COMPUTER SYSTEMS

Serial No. ‘ :

Filed: November 9, 1995 :' Greup-'

. PRELIMINARY AMENDMENT

Pittsburgh, Pennsylvania 15222

November 9, 1995

Hon. Commissioner of Patents
and Trademarks '

Washington, DC 20231

zSir:

Preliminary to the examination of the continuation

application filed herewith, please amend that application as

follows.

In the Claimg
3

Please can e1 claims 2 ~ 17,

“MA

93085c0N'X,Wé

REMARKS

This Preliminary Amendment is filed to reduce the number of claims for purposes of calculating the filing fee. Applicants intend to file a second preliminary amendment to either amend or replace claims 1 and 18 ~ 20.“

Respectfully submitted,

L. Pencoske

Registration No. 29,688

KIRKPATRICK & LOCKHART

1500 Oliver Building
Pittsburgh, PA 15222
(412) 355—8645

Attorney’s Docket No.WWW PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: James R. Woodhill et al.
Serial No.: 08 / 555,376 Group No.:
Filed: November 9 1995 Examiner:
For: SYSTEM AND METHOD FOR DISTRIBUTED WGEVENT ON

NETWORKED COMPUTER SYSTEMS ' ($7.3
Commissioner of Patents and Trademarks M {5‘91 £ij:
Washington, 0.0. 20231 ff “

AMENDMENT TRANSMITTAL ; l V.

CI)

1. Transmitted herewith is an amendment for this application.

STATUS

2. Appiicant is
E] a ema‘ti entity. A verified statement:
C] is attached.

C] was already filed.. FEB 2 0 $96
{33 other than a small entity.
GROUP 2?:hit)

CERTIFICATE OF MAiLRNGfl‘RANSMISSION (37 GFR 1.8a)
i hereby certify that this correspondence is. on the date shown beiow, being:
MAiLlNG FACSlMlLE

23 deposited with the United States Postai Cl transmitted by facsimile to the
Service with sufficient postage as first ciass Patent and Trademark Office
mail in an envelope addressed to the .. ,

Washington, Commissioner0.04of Patents 20231 and Trademarks, Q '
w, Signature
Cami: JAM V196 Edward L. Pencoske
(type or print name of person certifying)

(Amendment Transmittal [9~191—-page 1 of 4)

EXTENSION OF TERM

NOTE: “Extension of Time in Patent Cases (supplement Amendments) -—— If a timely and complete response has been filed after a Non-Final Ollice Action, an extension of time is not required to permit tiling and/or
entry 01 an additional amendment after expiration of the shortened statutory period.
If a timely response has been filed after a Final Office Action, an extension of time Is required to permit
filing and/or entry of a Notice of Appeal or filing and/or entry 0/ an additional amendment alter expiration
oi the shortened statutory period unless the timely~1iled response placed the application in condition
for allowance. Of course, if a Notice of Appeal has been flied within the shortened statutory period
the period has ceased to run " Notice of December 101985 (1061 O. G. 34-35).

NOTE: See 37 OFF? 1. 645 for extensions of time in Interference proceedings and 37 OFF? 1.550(0) for extensions of time in reexamination proceedings.

3. The proceedings herein are for a patent application and the provisions of 37 CFR
1.138 apply

(complete (a) or (b) as applicable)

(a) [:1 Applicant petitions for an extension of time under 37 CPR 1.136 (fees: 37 CFR
1.17(a)»(d) for the total number of months checked below:

Extension Fee for other than Fee for

(months) smali entity smali entity
:1 oneI month $ 110.00 35 55.00
:1 two months 39 370.00 $185.00
.. three months $ 870.00 $435.00
._. four months $1,360.00 $680.00

Fee 35

if an additional extension of time is required please consider this a petition therefor.

(check and complete the next item, if applicable)

1:1 An extension for

fee paid therefor 01$ is deducted from the total fee due for the total months of extension now requested.

Extension fee due with this request $
OR

(b) 13 Applicant believes that no extension of term is required. However this condi-

tional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition for extension of time.

(Amendment Transmittal [9-19]-—~page 2 of- 4)

months has already been secured and the.
z“

FEE FOR CLAIMS

4. The fee for claims (37 CFR 1.16(b)~(d)) has been calculated as shown below:
OTHER THAN A

(Col. 1) (Col, 2) (Col. 3) SMALL ENTITY SMALL ENTITY

CLAiMS

REMAtNiNG , HiGHEST No _ - AFTER PREVIOUSLY PRESENT ADDlT. ADDlT.
AMENDMENT PAlD FOR EXTRA RATE FEE on RATE FEE

TOTAL ' 18 MINUS " 20 = 0 x11; $ 0- x22== $ 0

INDEP. ' 2 ' MINUS 3 = O x38= s 0 x76= $ 0

Ci FIRST PRESENTATION OF MULTIPLE DEP. CLAlM +129:,~$ +240: s

TOTAL on TOTAL 0
AOOIT. FEE e ADDIT.
FEE e

' If the entry in Col. 1 is less than entry in Col. 2, write “0" in Col. 3. '
" if the “Highest No. Previously Paid for" W THIS SPACE is less than 20. enter "20".
If the “Highest No, Previously Paid For" lN THIS SPACE is less than 3. enter “3".
The “Highest No. Previously Paid For" (Total or indep.) is the highest number found in the appropriate box in Col. 1 of a prior amendment or the number of claims originally filed.

WARHIHG: “After final rejection or action (§ 1.1 13) amendments may be made cancelling claims or complying with-'any requirement of form which has been made.” 37 CFR § 1.116(5) (emphasis added).

{compiete (c) or (d) as applicable)

(c) E} No additional fee for claims is refluired.

0R

(d) C] Total additional fee for claims required 3

FEE PAYMENT

[3 Attached is a check in the sum Of $WW.

Ci Charge Account No. W the sum of
S .

A duplicate of this transmittal is attached.

(Amendment Transmittal [9-19]—-~page 3 of 4)

FEE DEFICIENCY

NOTE: if there is a tee deficiency and there is no authorization to charge an account: additional fees are necessary to cover the additional time consumed in making up the original deficiency. It the maximum, six-month period has expired before the deficiency is noted and corrected, the application is held abandoned. In those instances where “authorization to charge is included, processing delays are encountered in returning the papers to the PTO Finance Branch in order to apply these charges prior
to action on the cases. Authorization to charge the deposit account for any fee deficiency should be checked. See the Notice of April 7, 1986, (1065 0.6. 31—33).

6. E] if lanylaijdgtional - I extension and/or fee is required, charge Account No. t

ANDIOR

Ea if any additional fee for claims is required, charge

Account No..__L1:,.LLLO._*.,_.___._._..

SIGNATURE OF ATTORNEY

Reg,No.: 29,688
Edward L. Pencoske

type or print name of attorney

TELNQ'ICIIQ) 355*8645 >
‘ Kirkpatricke Lockhart LLP

‘P.o.Address 1500 Oliver Building
Pittsburgh, PA 15222

(Amendment Transmittal [9-19lmpage 4 of 4)

E
i
i

(‘3, “'V VI»
a? COPY ‘3)”

g,.

PATENT

’)’¢" b 93085CON

w30q
IN THE UNITED STATES PATENT AND TRADEMARK OE?E§§1Q
idtmcaaad

Art Unit a ‘ W90 FEB 2£3
Examiner : GROUP 23300
In re application of : SYSTEM AND METHOD FOR
DISTRIBUTED MANAGEMENT ON
James R. oodhill et al. : 'NETWORKED COMPUTER SYSTEMS
0‘ x
Serial NO. 555,376

Filed: November 9, 1995 : Group

SECOND PRELIMINARY AMENDMENT

")

Pittsburgh, Pennsylvania: E5222

f 5 3‘ all.“

January 5, 1996 gs in

, Hon. Commissioner of Patents ‘ 39 m HQ

and Trademarks ‘,HV $9
Washington, DC 20231 - é?

Sir:

Preliminary to the examination of the application filed

November 9, 1995, please amend that application as follows.

‘In the Claims

Please cancel claims 1 and 18 — 20, which are all of the

claims remaining in the application, and substitute the following new claims.

—~21. A system for distributed management of the storage
space and data on a networked computer system wherein the networked computer system includes at least two storage deyices for storing data files, said distributed storage management
system comprising:

Emeans for selectively copying data files stored on one of
the storage devices to another of the storage devices;

y

,

i
i

means for dividing each data file into one or more binary objects of a predetermined size;
means for calculating a current value for a binary object

identifier for each binary object within a file, said
calculation of said binary object identifier being based upon hhfl

the actual data contents of the associated binary object, said

calculated binary object identifier being saved as the name of

the associated binary object; I means for comparing said current name of a particular
binary object to one or more previous names of said binary_

object; ‘ means for storing said current name of said binary object
identifier; and
means for controlling said means for selectively copying
binary objects in response to said means for comparing.

22. The distributed storage management system of claim 21
wherein said means for calculating said current name includes means for calculating a current name compriSed of at least two independently determined values.

23. The distributed storage management system of claim 21_
wherein said means for calculating said current name includes means for calculating a 128-bit binary value comprised of four 32—bit fields.

24. The dietributed storage management system of claim 23
‘wherein said four 327bit fields include a binary object identifier size field, a Cyclical Redundancy Check number field

calculated against the contents of the binary object, a Longitudinal Redundancy Check number field calculated against the contents of the binary object, and a binary object hash_

number field calculated against the contents of the binary

object.

25. The distributed-storage management system of claim 21

further including means for auditing the performance of said distributed storage management system, said means for auditing including:

second means for controlling said means for selectively_ copying binary objects to recopy a previously copied binary Object;
means for recalculating said binary object identifier for
the recopied binary object, said recalculated binary object identifier being saved as the name of the associated binary

object; ‘ ‘ means for comparing said recalculated name to a previous
name of said binary object; and

means for reporting a failure if said recalculated name is
not identical to said previous name of said binary object.

26. The distributed storage management system of claim.21

wherein said means for controlling said means for selectively copying includes means for instructing, in response to said
means for comparing, said means for selectively copying to copy

a particular binary object only if its current name is not identical to a previous name for that particular binary object.

27. The distributed storage management system of claim 21
additionally comprising means for segmenting the binary objects into granules of data, and wherein said granules of data are processed in the same manner as said binary objects.

28. The distributed storage management system of claim 27 further including means for reconstructing a binary object from a most recent complete copy of the binary object, said means for reconstructing including:
means for copying said granules copied by said means for

selectively Copying granules to said most recent complete copy of the binary object in order from most-recently copied granule

to least-recently copied granule; and
means for generating a bitmap for controlling said means
for copying said copied granules.

29. The distributed storage management system of claim 28 wherein said means for calculating said current name for said granule includes means for calculating a 32~bit Cyclical Redundancy Check number calculated against the contents of said granule and means for calculating a 32—bit binary object hash number calculated against the contents of said granule.

30./ The distributed storage management system of claim 28 further including means for restoring a current version of a binary object to a previous versiOn of that binary object, said means for restoring including:

means for calculating said name for.each of said granules

in the current version of the binary object;

means for comparing said calculated name to a previous name
for each of said granules in the current version of the binary object; and

means, responsive to said means‘for comparing said names,
for replacing those granules in the current version of the binary object for which said names arelnot identical.

31. The distributed storage management system of claim 21
additionally comprising;

means for indicating which of said copied binary objects

must be copied to a particular storage device in the event of a failure of that storage device before that storage device is considered to be operable by the local computer with which that storage device is in communication; and
wherein said means for controlling said means for
selectively copying binary objects is further responsive to said
means for indicating.

32. The distributed storage management system of claim 31
v wherein said means for indicating includes means for specifying a last access date such that only binary objects that have been accessed by the networked computer system on or after said last access date must be copied to a particular storage device.

before

that storage device is considered to be operable.

33. The distributed storage management system of claim 21
additionally comprising:
means for maintaining a file retention list wherein said
file retention list includes a file retention pattern for each

binary object copied by said means for selectively copying

binary objects;

means for determining which of the binary objects copied by said means for selectively copying binary objects match each of

said file retention patterns; and
means for deleting the binary objects from the storage
devices in response to said means for determining. , ‘ ‘

34. The distributed storage management system of claim 33

wherein said file retention pattern includes daily, weeklyk monthly, quarterly and yearly retention patterns.

35. The distributed storage management system of claim 2i.

wherein said networked computer system includes a remote backup file server, and wherein said means for selectively copying copies the binary objects stored on one of the storage devices to another of the storage devices or to the remote backup file server, said distributed storage management system additionally comprising:
means for employing user—defined priorities to determine
which binary objects are to be copied to another storage device and to determine a queuing sequence for copying binary objects to the remote backup file server.

36. A method for management of the storage space and data
on a computer system wherein the computer system includes at
least two storage areas for storing data files, said method
comprising the steps of:

dividing each data file into one or more binary objects of
a predetermined size; V
calculating a current value for a binary object identifier
for each binary object within a file, said calculation of said binary object identifier being based upon the actual data

'contents of the associated binary object, said calculated binary object identifier being saved as the name of the associated Ibinary object;
comparing said current name of said binary object to one or
more previous names of said binary object;

storing said current name of said binary object as a
previous name of said binary object; and

selectiyely copying binary objects in response to said comparing step.

’37. The method of claim 36 wherein said step‘of calculating said current name for said binary object includes the step of calculating a current name for a binary object comprised of at least two independently determined values.

38. The method of claim 36 wherein said step of

calculating said current name for said binary object includes the step of calculating a current name for a binary object‘

utilizing a 128nbit binary value comprised of four 32~bit fields and wherein said four 32—bit fields include a binary object

identifier size field, a Cyclical Redundancy Check number field calculated against the contents of the binary object, a Longitudinal Redundancy Check number field calculated against the contents of the binary object, and a binary object hash number field calculated against the contents of the binary ,object.——

REMARKS
This Second Preliminary Amendment is filed to submit new
claims for examination.

In the parent application, of which the instant application
is a continuation, an Office Action was issued on 11 May 1995

rejecting claims 1 — 20 under 35 U.S.C. § 103 as being

unpatentable oVer an article to Daniel Barbara et al. ("the
Barbara article"). It is the Examiner’s position, as set forth
on page 3 of the Office Action in the parent application, that

— the Barbara article discloses "a means for storing".. As the Examiner noted, the Barbara article did not specifically]

. I
calculate the check sums to be used as identifiers for selected

1 ‘I ~

5$UUDLQUN
IN THE UNITED STATES.?ATENT AND TRADEMARK OFFICE
“£9

Art Unit \ o “7‘9Ab
Examiner

In re application of. SYSTEM AND METHOD FOR
DISTRIBUTED MANAGEMENT ON

James R. Woodhill et a1.' NBTWORKED COMPUTER SYSTEMS
Serial No. 555,376
Filed: November 9, 1995 Group

gECOND PRELIMIHAB! AMEQQMgNT

‘?itteburgh, Pennsylvania lSZéz

January 5, 1996

Hon. Commissioner of Patents ': F%W€MEWXfEE3
and Trademarks 0PT(9€/W96
@W‘Qi ”:3 QQQQ

Washington, DC 20231
Sir:

Preliminary to the examination of the application filed November 9, 1995, pleaee amend that application as follows.

In the C ’

Please cancel c ims l and 18 ~ 20, which are all of the
claims remaining in the “ lioation, and substitute the
following new claims.

“figié‘ A system for distributed management of the storage
space and data on a networked computer system wherein the networked computer system includes at least two storage devices
for storing data files, said distributed storage management
system comprising:
means for selectively copying data files stored on one of
the storage devices to another of the storage devices,

PX 1-5667“.1

wherein said means for calculating said current name includes

means for calculating a current value for a binary object

identifier for each binary object within a file, said calculation of said binary object identifier being based upon

vthe actual data contents of the associated binary object, said

calculated binary object identifier being saved as the name of the associated binary object;
means for comparing~said current name of a particular
'binary object to one or more previous names of said binary object;
means for storing said current name of said binary object identifier; 0 and
means for controlling said means for selectively copying
binary objects in response to eaid'means for comparing.

921 , ail 'The distributed storage management system of claimlzi
wherein said means for calculating said current name includes

means for calculating a current name comprised of at least two ' I independently determined values,

if“ k ‘ ‘ i - /
. The distributed storage management system of claim/2f

means for calculating a 128~ bit binary value comprised of four
32bit fields

7/ ‘.' >.‘ ' _ V ' v ‘3 ' 241 The distributed storage management system of clain1}%{
wherein said four 32-bit fields include a binary object

identifier size field{ a Cyclical Redundancy dheck number field calculated against the contents of the binary object, a

Longitudinal Redundancy Check number field calculated against the contents of the binary object, and a binary object hash

l

j” i '

, /
;y{.- The distributed storage management system of claim/Z5 further including means ior.auditing the performance of said distributed storage management system, said means for auditing

including:
second means for controlling said means for selectively

copying binary objects to recopy a previously copied binary object; V ,»
means for recalculating said binary object identifier for
‘ the recopied binary object, said recalculated binary object

identifier being saved as the name of the associated binary

object; ’ means for comparing said recalculated name to a previous
name of said binary object; and

means for reporting a failure if said recalculated name is

.not identical to said previous name of said binary object.

(p - I /

96: The distributed storage management system of claim/}(
wherein said means for controlling said means for selectively copying includes means for instructing, in response to said
means for comparing, said means for selectively copying to copy

a particular binary object only if its current name is not identical to a previous name for that particular binary object.

/

of. The distributed storage management system of claim zd/__ additibnally comprising means for segmenting the binary objectsH. into granules of data, and wherein said granules of data are ‘proceseed in the same manner as said binary objects.

I
I

to least—recently copied granule; and

. binary object to a previous version of that binary object, said

' object; and

a most recent complete copy oi the binary object, said means for, reconstructing including:
means for copying said granules copied by said means for selectively copying granules to said most recent complete copy‘
of the binary object in order from moat-recently copied granule

means for generating a bitmap for controlling said means
for copying said copied granules.

. 8

;Q< The distributed storage management system of claim/g6,
wherein said means for calculating said current name for said granule includes means for calculating a szebit Cyclical

Redundancy Check number calculated against the contents of said granule and means for calculating a 32~bit binary object hash

number calculated against the contents of said granule. ‘ l
/0 8

/}5. The distributed storage management system of claim ga/
further including means for restoring a current version of a

means for restoring including:
means for calculating said name for each of said granules
in the current version of the binary objectx'
means for comparing said calculated name to a previous name' for each of said granules in.the current version of the binary

means, responsive to said means for comparing said names.
for replacing those granules in the current version of the binary object for which said names are not identical;

means Lox: lnuiuntuzg wuiuu UL caiu uuyiuu uiueiy UUJQULD

must be copied to a particular storage device in the event of a failure of that storage device before that storage device is

considered to be operable by the local computer with which.that storage device is in communication; and ' Lwherein said means for controlling said means for selectively copying binary objects is further responsive to said.‘

means for indicating.

/ ‘ ' 1/.

;Z< The distributed storage management system of claim/yi
wherein said means tor indicating includes meens tor specifying a last access date such that only binary objects that have been accessed by the networked computer system on or after said last

access date must be copied to a partibular storage device before that storage device is considered to be operable.

/ V. , igéi. The distributed.etoragc management system of claim:;{
additionally comprising:
means for maintaining a file retention list wherein said

file retention list includes a file retention pattern for each binary object copied by said means for selectively copying binary objects; .

means for determining which of the binary objects copied by said means for selectively copying binary objects match each of said file retention patterns} and

means for deleting the binary objects from the storage
devices in response to said means for determining.

monthly, quarterly and yearly retention patterns.

l5;a{. The distributed storage management system of claim/;%
wherein said networked computer system includes a remote backup. file server, and'wherein said means for selectively copying

copies the binary objects stored on one of the storage devices

to another of the storage devices or to the remote backup file server, said distributed storage management system additionally comprising:

a means for employing user-defined priorities to determine which binary objects are to be copied to another storage device ‘

and to determine a queuing sequence for copying binary objects
to the remote backup file server.

lféé. A method for management of the storage Space and data
on a computer system wherein the computer system includes at V least two storage areas for storing data files, said method_i§i comprising the steps of: i ’
dividing each data file into one or more binary objects of
a predetermined size; ‘
Calculating a current value for a binary object identifier
for each binary object within a file, said calculation of said binary object identifier being based upon the actual data
1 contents or the associated binary object, said calculated binary‘ object identitier being saved as the name of the associated

binary object;

I comparing said current name of said binary object to one or
, more previous names of said binary object;

storing said current name of said binary object as a
preVious name of said binary object; and

comparing Step. I a . . ' i

1/7 / , §fi< The method of claim }é}:herein said step of”
calculating said current name for said binary object includes
the step of calculating a current name fer a binary object comprised of at leaat two independently determined values.

A3 ~ / 3

}K{ The method of claim 6 wherein said step of
calculating said-current name for said binary object includes

tna step of calculating_a current name for a binary object utilizing a 128—bit binary value ccmnriaed of four 32-bit fields and wherein said four 32-bit fielda include a binary object identifier size field, a Cyclical RedUndancy Check number field calculated against the contents of the binary object, a

Longitudinal Redundancy Check number field calculated against the contents of the binary object, and a binary object hash number field calculated against the contents of the binary

cbject.--

REMARKS

This Second Preliminary Amendment is filed to submit-new

claims for examination.

In the parent application, cf which the instant application
is a continuation; an Office Acticn was issued on 11 May 1935

rejecting claims 1 - 20 uncer 35 U.S.C. 5 103 as being unpatentable over an article to Daniel Barbara et al. ("the Barbara article"). It is the Examiner's poeition,‘ae set forth on page 3 of the Office Action in the parent anplication, that the Barbara article discloses "a means for storing"; As the Examiner noted, the Barbara article did not specifically calculate the check sums to be used as identifiers tor selected

- 7 _

SERIAL NUMBER FILING DATE 1 ‘ FIRST NAMED APPLICANT ’ ATTORNEY DOCKET NOJ ‘

08/555, 1376 1 1 I 09/95 WDCIDHILL Al_mm-.,,33fl8tfiflbflww.
EWWJEWWW; I

E3M1/01fl2

eownno L PENCDSKE []f§g§§§gfft";aggfiam§;fj

KIRKPATRICK e LDCKHART ' *‘ “”" .,:““*
1500 OLIVER BUILDING . I5

PITTSBURGH PA 15222 2307

DATE MAILED: {lung/9'7

NOTICE OF ALLOWABILITY

PART I. . '

1.4K Thiscommunication is responsive to ” L- . 02 79
2. 5% All the claims being allowable, PROSECUTION ON THE MERITS l8 (OR REMAINSI=CLOSED in this application It not included herewith (or previously mailed). a Notice OI Allowance And issuetFeeDue or other appropriate communication will be sent in due

course.

33’

4. I:

The
The drawings

allowed claims

tiled on

are
W

MQLJMLMMMWLW
are acceptable.
5. CI Acknowtedgment IS made‘ot the claim for priority under 35 U,S.C. 119. The certified copy has I-) been received. I-) not been received, i__] been filed in parent application Serial No. . filed on 4
6. I: Note the attached Examiner‘s Amendment.’
7. I: Note the attached Examiner Interview Summary Record. PTOLuIiS,
ELK/Note the attached Examiner's'Statement oi Reasons for Allowance.
9. I: Note the attached NOTICE OF REFERENCES CITED. PTO-892.
to. C] Note the attached INFORMATION DISCLOSURE CITATION. PTO-1449.

PART II.

A SHORTENED STATUTORY PERIOD FOR RESPONSE to comply with the requirements noted below is set to EXPIRE THREE MONTHS FROM THE "DATE MAILED" indicated on this lorm, Failure to timeiy comply will result in the ABANDONMENT oi this application.- ExtenSlons oi time may be obtained under the provisions of 37 CFR 1.t36(a),

t. [:3 Note the attached EXAMINER‘S AMENDMENT or NOTICE OF INFORMAL APPLICATION, PTO-152. which discloses that the oath or declaration is deilcient. A SUBSTITUTE OATH OR DECLARATION lS REQUIRED.

2,EKAEPLICANT MUST MAKE THE DRAWING CHANGES INDICATED BELOW IN THE MANNER SET FORTH ON THE REVERSE SIDE OF THIS PAPER.

am Dran 9 mtormaltties are indicated on the NOTICE RE PATENT DRAWINGS. PTOdeS. attached hereto or to Paperl No.

. CORRECTION IS REQUIRED

b. [3 The proposed drawing ‘correction tiled on
REQUIRED.

c C] Approved dlawing corrections are described by the examiner in the attached EXAMINER‘S AMENDMENT, CORRECTION is

has been approved by the examineri CORRECTION IS

REQUIRED.

(3261 Format drawrngs are now REQUIRED.
_...._......._.__..__................._~........._.._...._-_....—u...~__..~.......—,.. ————————— u.-_.....~———~_—_..~u.__..-——~.....-...............-__—-........._H_ _________

Any response to this letter should include in the upper right hartd corner. the following iniormation Irom the NOTICE OF ALLOWANCE , AND iSSUE FEE DUE. iSSUE BATCH NUMBER, DATE OF THE NOTICE OF ALLOWANCE AND SERIAL NUMBER, 3" “ l
Attachments:

4_ Examiner 5 Amendment .. Notice oi informal Application. PTO~ i5?
, Examiner Interview Summary Record. PTOL- 413 ‘ - Notice re Patent Drawings. PTO-948
g/Fieasons lor All0wance _ Listlng oi Bonded Drattsmen
* Notice ol Reierences Citedi PTO~892 _ Other
._ lnlormatlon Disclosure Citation. PTO-1449. '