

Worksheet 3 – Copyright Damian Brasher www.diap.org.uk 26th May 2008 FDL**Page 1**

This sheet describes the architecture of the DIAP system starting with the directory / storage slot layout followed by the copies described as cron jobs, this is how they appear in the bash script prototype.

Page 2-3

There are descriptions of the mechanisms in place and these are followed by a breakdown of the calculations then a simple worked example.

This basic structure has take about a year to develop, I don't envisage any more major changes although some details may be added. See design philosophy on main website.

Directory layout

Slots (dirs)	A	
Monthly FULL from bkp tool	aFull01	
Last months FULL	aFull02	
Daily Diff from a bkp tool	ad00	
A	B	C
aFull01	bFull01	cFull01
aFull02	bFull02	cFull02
ad00		
ad01	bd01	cd01
ad02	bd02	cd02
ad03	bd03	cd03
ad04	bd04	cd04
ad05	bd05	cd05
ad06	bd06	cd06
ad07	bd07	cd07
ad08	bd08	cd08
ad09	bd09	cd09
ad10	bd10	cd10
ad11	bd11	cd11
ad12	bd12	cd12
ad13	bd13	cd13
ad14	bd14	cd14
ad15	bd15	cd15
ad16	bd16	cd16
ad17	bd17	cd17
ad18	bd18	cd18
ad19	bd19	cd19
ad20	bd20	cd20
ad21	bd21	cd21
ad22	bd22	cd22
ad23	bd23	cd23
ad24	bd24	cd24
ad25	bd25	cd25
ad26	bd26	cd26
ad27	bd27	cd27
ad28	bd28	cd28
ad29	bd29	cd29

NB: The model assumes that differentials are always smaller than FULL copies.

Copyright (c) 2008 Damian Brasher. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
<http://www.gnu.org/licenses/fdl-1.2.txt>

Cont next page...

Cron jobs

Node

Node

Sheet1

Day – Time	A	B	Day – Time	A	B
D1 – T=0	aFull01->bFull01		D11 – T=0	ad00->ad10	
D2 – T=0		bFull01->cFull01	D11 – T=0	ad00->bd10	
D2 – T=0	ad00->ad01		D11 – T=3		bd10->cd10
D2 – T=0	ad00->bd01		D12 – T=0	ad00->ad11	
D2 – T=3		bd01->cd01	D12 – T=0	ad00->bd11	
D3 – T=0	ad00->ad02		D12 – T=3		bd11->cd11
D3 – T=0	ad00->bd02		D13 – T=0	ad00->ad12	
D3 – T=3		bd02->cd02	D13 – T=0	ad00->bd12	
D4 – T=0	ad00->ad03		D13 – T=3		bd12->cd12
D4 – T=0	ad00->bd03		D14 – T=0	ad00->ad13	
D4 – T=3		bd03->cd03	D14 – T=0	ad00->bd13	
D5 – T=0	ad00->ad04		D14 – T=3		bd13->cd13
D5 – T=0	ad00->bd04		D15 – T=0	ad00->ad14	
D5 – T=3		bd04->cd04	D15 – T=0	ad00->bd14	
D6 – T=0	ad00->ad05		D15 – T=3		bd14->cd14
D6 – T=0	ad00->bd05		D16 – T=0	ad00->ad15	
D6 – T=3		bd05->cd05	D16 – T=0	ad00->bd15	
D7 – T=0	ad00->ad06		D16 – T=3		bd15->cd15
D7 – T=0	ad00->bd06		D17 – T=0	ad00->ad16	
D7 – T=3		bd06->cd06	D17 – T=0	ad00->bd16	
D8 – T=0	ad00->ad07		D17 – T=3		bd16->cd16
D8 – T=0	ad00->bd07		D18 – T=0	ad00->ad17	
D8 – T=3		bd07->cd07	D18 – T=0	ad00->bd17	
D9 – T=0	ad00->ad08		D18 – T=3		bd17->cd17
D9 – T=0	ad00->bd08		D19 – T=0	ad00->ad18	
D9 – T=3		bd08->cd08	D19 – T=0	ad00->bd18	
D10 – T=0	ad00->ad09		D19 – T=3		bd18->cd18
D10 – T=0	ad00->bd09		D20 – T=0	ad00->ad19	
D10 – T=3		bd09->cd09	D20 – T=0	ad00->bd19	
			D20 – T=3		bd19->cd19

Cont next page...

Node

Sheet1

Day – Time	A	B
D21 – T=0	ad00->ad20	
D21 – T=0	ad00->bd20	
D21 – T=3		bd20->cd20
D22 – T=0	ad00->ad21	
D22 – T=0	ad00->bd21	
D22 – T=3		bd21->cd21
D23 – T=0	ad00->ad22	
D23 – T=0	ad00->bd22	
D23 – T=3		bd22->cd22
D24 – T=0	ad00->ad23	
D24 – T=0	ad00->bd23	
D24 – T=3		bd23->cd23
D25 – T=0	ad00->ad24	
D25 – T=0	ad00->bd24	
D25 – T=3		bd24->cd24
D26 – T=0	ad00->ad25	
D26 – T=0	ad00->bd25	
D26 – T=3		bd25->cd25
D27 – T=0	ad00->ad26	
D27 – T=0	ad00->bd26	
D27 – T=3		bd26->cd26
D28 – T=0	ad00->ad27	
D28 – T=0	ad00->bd27	
D28 – T=3		bd27->cd27
D29 – T=0	ad00->ad28	
D29 – T=0	ad00->bd28	
D29 – T=3		bd28->cd28
D30 – T=0	aFull01->aFull02	
D30 – T=0		bFull01->bFull02
D30 – T=0		cFull01->cFull02
D30 – T=0	ad00->ad29	
D30 – T=0	ad00->bd29	
D30 – T=3		bd29->cd29

Start 00:00

End 00:06

T = 0 (00:00)

T = 3 (03:00)

All copies from a-a are not used in bandwidth

requirement calculations as internal copies are fast enough to ignore for prototype purposes.

Two entry points, aFull01 beginning of month and ad00 for the remaining days. Assuming entry points are filled during the day before the cycle begins at night. cron jobs split between 3 nodes, ad00 is cleared after copy to bd\$

The system reduces single point of failure by creating a single FULL copy on each node at the beginning of the month then at the end of the month to cover the next 30 day diap cycle. Storing the differentials in a distributed manner. Use a program such as Bacula set to use a monthly cycle.

The copies between a-a and a-b occur in the first three hours then the copy from b-c happens after three hours. These times can be changed as required.

Because the last nightly copy occurs between b and c, use of node c is optional. Nightly copies to and between nodes are made to new directories, if due to some failure a node is unavailable then the copy is not made, however the next day when communication is restored copies continue to the next nightly directory. This increases robustness over previous layout as the next nightly copy is not dependent on the success of the previous night's copy. Only two copies between nodes are made between days 3-30, day 1 a single FULL and day 2 FULL and Day 30 does make an internal copy on all nodes.

Calculations:-

LBM = Lowest Maximum Bandwidth between any three nodes

NB: actual max transfer will vary so test transfers are recommended for accuracy

LBM assumes all available bandwidth is allocated to running DIAP.

Max aFULL01 = LMB x 6 hrs

This assumes no transfer interruptions and that the maximum bandwidth is constant.

Ave. Diff = (Sum 29 (or a month) Daily Differentials) / 29

Ave Differential is variable depending on your storage growth, this represents a trend and can be an estimate to start with, but by watching the trend of Differential growth more accurate calculations can be made. It is assumed your Differentials are always less than the initial FULL copy.

Min DIAP Dir size (node a) = (Max aFULL01 x 2) + (29 x Ave. Diff) + (1 x Ave. Diff)

Plus 1 x Ave. Diff to account for ad0

Min DIAP Dir size (node b/c) = (Max aFULL01 x 2) + (29 x Ave. Diff)

You can include transfer log files in the Min DIAP Dir size, for simplicity they have been omitted.

Example system:	LMB x 6 hrs	Ave. Diff	Max aFULL01
LBM occurs between b->c	1Mbit/Sec	Est. 500 MiB	2.6 GiB

Min DIAP Dir size (node a)

$(2.6 \times 2) + (29 \times 0.5) + 0.5 = 20.2 \text{ GiB}$

Min DIAP Dir size (node b-c)

$(2.6 \times 2) + (29 \times 0.5) = 19.7 \text{ GiB}$

DIAP Dir: this is the working directory used on each node and contains all DIAP configuration working and storage directories.

If a copy fails then the system will retry the next day but you lose the day of failure.

Using rsync logs you can trace / track the successful copies