

Some information about UC5

The text of UC5:

“Like [UC2], but AU grows above its max promised size. Must indicate a new max space, determine whether (a) at least k hosts already harvesting AU have sufficient resources for new size; (b) if not a, then can additional hosts in the network be designated to harvest AU, and some existing hosts stop harvesting *while retaining older files*; (c) if not (b) minimum #/size hosts that would have to be added.”

UC5 is so complex that I broke it up into 4 different parts. The initial setup is each LOCKSS box has been invited to harvest a certain set of AU's. This initial setup is indicated in the diagram for AU0, but is not the same setup as would be found in another use case. The diagrams themselves are cumulative, e.g. the diagram for AU2 reflects the operation of the AU0 and AU1 diagrams, and so forth. Each diagram shows what happens when each of the AU's exceeds its promised size and announce a new max size. Maybe it is not reasonable in real life that they know what their new max sizes will be, but use case says that a new max space is indicated.

I assume during this case that all changes happen in a linear fashion, i.e., they happen when I say they happen and in the order that I say they happen. This assumption is about as ludicrous as a statement that I can stop a hard drive failure by saying “Fail not.” But I cannot show a real model of what happens in a nondeterministic system without simulation (and certainly not easily on a sheet of paper).

I further assume that when the allocated space for harvesting a particular AU is exceeded, the LOCKSS box will stop harvesting at that point, and determine whether it continues to have sufficient committed space. If it does not, then it simply stops harvesting (I use the label “deinvited”). It does not delete its progress, but it does not overrun its allocated space for the AU if unallocated committed space is insufficient. This assumption may too be poor; a box may not know enough to make these determinations. However, this diagram can easily be altered to better reflect a change in this assumption.

The LOCKSS boxes and AU's on a diagram reflect the state of the system at the instant when an AU's maximum promised size increases. The operation of this use case occurs within the schema server box, and its results are reflected in the diagram for the next AU.

I am very aware that these diagrams probably may not make much sense, and so if more explanation or another type of diagram is desired, please let me know.

Data-PASS SSP Use Case 5 AU0

lockss-0.icpsr.umich.edu

IP Add 141.211.192.29
Commitment 100



lockss-1.icpsr.umich.edu

IP Add 141.211.192.52
Commitment 1700



lockss-2.icpsr.umich.edu

IP Add 141.211.192.53
Commitment 1700



props.irss.unc.edu

IP Add 152.2.32.234
Commitment 39



dris.irss.unc.edu

IP Add 152.2.32.233
Commitment 39



haar.irss.unc.edu

IP Add 152.2.32.207
Commitment 200



Machine Ownership Key	
	ICPSR
	Odum Institute

Invitation AU0, AU3

Invitation AU0, AU1, AU2, AU3

Invitation AU0, AU1, AU2, AU3

Invitation AU1, AU2, AU3

Invitation AU1, AU2

Invitation AU0

SCHEME UC5

Let there be $h=6$ hosts and $n=4$ AU's. Let invitations be previously assigned accepted as follows:

When an AU exceeds its promised size, a new maximum size must indicated. Then generate new invitations such that:

(A) For each host, determine if committed space is sufficient without displacing other AU's harvested by that host. If a host has sufficient committed space, it increases allocated space to new size, and continues harvesting. No new invitation change needed if $k=4$ hosts are still harvesting.

(B) If a host does not have sufficient committed space, it stops harvesting the AU while retaining old files. New invitations are issued to bring $k=4$ hosts harvesting each AU, without displacing any other current assignments or deleting files.

(C) If $k=4$ hosts cannot be found to harvest an AU, determine the minimum number and size of hosts that would have to be added.

AU0 {lockss-0, lockss-1, lockss-2, haar}
Promised: 50
New max size: 170

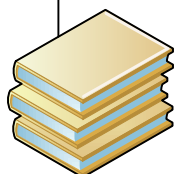
- (A) lockss-0: Harvesting 54, committed 100. Cannot handle increase. $k=3$.
- lockss-1: Harvesting 84, committed 1700. Can handle increase. Increase allocation, harvesting 204, committed 1700. $k=3$.
- lockss-2: Harvesting 84, committed 1700. Can handle increase. Increase allocation, harvesting 204, committed 1700. $k=3$.
- haar: Harvesting 50, committed 200. Can handle increase. Increase allocation, harvesting 170, committed 200. $k=3$.

Fewer than $k=4$ hosts are harvesting AU0.

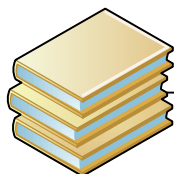
- (B) *props has insufficient space (harvesting 34, committed 39).
- *dris has insufficient space (harvesting 30, committed 39).

No nonharvesting hosts available.

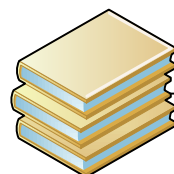
(C) 1 more host is necessary, which must be able to harvest 170.



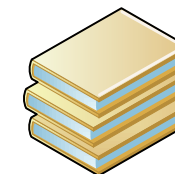
Title	AU0
Size	50
New Max Size	170



Title	AU1
Size	15



Title	AU2
Size	15



Title	AU3
Size	4

Text at <https://wiki.hmdc.harvard.edu/DPP/index.php?pagename=DPP.UseCases>

Data-PASS SSP Use Case 5 AU1

lockss-0.icpsr.umich.edu

IP Add 141.211.192.29
Commitment 100



lockss-1.icpsr.umich.edu

IP Add 141.211.192.52
Commitment 1700



lockss-2.icpsr.umich.edu

IP Add 141.211.192.53
Commitment 1700



props.irss.unc.edu

IP Add 152.2.32.234
Commitment 39



dris.irss.unc.edu

IP Add 152.2.32.233
Commitment 39



haar.irss.unc.edu

IP Add 152.2.32.207
Commitment 200



Machine Ownership Key	
Yellow	ICPSR
Blue	Odum Institute

Invitation	AU3
Newly Invited	0
Deinvited	AU0

Invitation	AU0, AU1, AU2, AU3
Newly Invited	0
Deinvited	0

Invitation	AU0, AU1, AU2, AU3
Newly Invited	0
Deinvited	0

Invitation	AU1, AU2, AU3
Newly Invited	0
Deinvited	0

Invitation	AU1, AU2
Newly Invited	0
Deinvited	0

Invitation	AU0
Newly Invited	
Deinvited	

SCHEME UC5

Let there be $h=6$ hosts and $n=4$ AU's. Let invitations be previously assigned accepted as follows:

When an AU exceeds its promised size, a new maximum size must indicated. Then generate new invitations such that:

(A) For each host, determine if committed space is sufficient without displacing other AU's harvested by that host. If a host has sufficient committed space, it increases allocated space to new size, and continues harvesting. No new invitation change needed if $k=4$ hosts are still harvesting.

(B) If a host does not have sufficient committed space, it stops harvesting the AU while retaining old files. New invitations are issued to bring $k=4$ hosts harvesting each AU, without displacing any other current assignments or deleting files.

(C) If $k=4$ hosts cannot be found to harvest an AU, determine the minimum number and size of hosts that would have to be added.

AU1 {lockss-1, lockss-2, props, dris}

Promised: 15

New max size: 21

(A)

lockss-1: Harvesting 204, committed 1700.

Can handle increase.

Increase allocation, harvesting 210, committed 1700. $k=4$.

lockss-2: Harvesting 204, committed 1700.

Can handle increase.

Increase allocation, harvesting 210, committed 1700. $k=4$.

props: Harvesting 34, committed 39.

Cannot handle increase. $k=3$.

dris: Harvesting 30, committed 39.

Can handle increase.

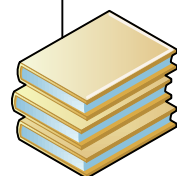
Increase allocation, harvesting 36, committed 39. $k=3$.

Fewer than $k=4$ hosts are harvesting AU1.

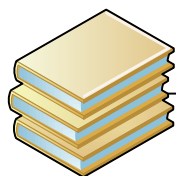
(B)

*lockss-0 is harvesting 54, committed 100.

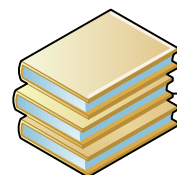
Invite lockss-0 (harvesting 75, committed 100). $k=4$.



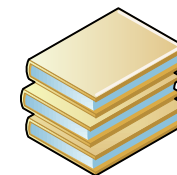
Title	AU0
Size	170



Title	AU1
Size	15
New Max Size	21



Title	AU2
Size	15



Title	AU3
Size	4

Text at <https://wiki.hmdc.harvard.edu/DPP/index.php?pagename=DPP.UseCases>

Data-PASS SSP Use Case 5 AU2

lockss-0.icpsr.umich.edu

IP Add 141.211.192.29
Commitment 100



lockss-1.icpsr.umich.edu

IP Add 141.211.192.52
Commitment 1700



lockss-2.icpsr.umich.edu

IP Add 141.211.192.53
Commitment 1700



props.irss.unc.edu

IP Add 152.2.32.234
Commitment 39



dris.irss.unc.edu

IP Add 152.2.32.233
Commitment 39



haar.irss.unc.edu

IP Add 152.2.32.207
Commitment 200



Machine Ownership Key	
Yellow	ICPSR
Blue	Odum Institute

Invitation	AU3
Newly Invited	AU1
Deinvited	AU0

Invitation	AU0, AU1, AU2, AU3
Newly Invited	0
Deinvited	0

Invitation	AU0, AU1, AU2, AU3
Newly Invited	0
Deinvited	0

Invitation	AU2, AU3
Newly Invited	0
Deinvited	AU1

Invitation	AU1, AU2
Newly Invited	0
Deinvited	0

Invitation	AU0
Newly Invited	0
Deinvited	0

SCHEME UC5

Let there be $h=6$ hosts and $n=4$ AU's. Let invitations be previously assigned accepted as follows:

When an AU exceeds its promised size, a new maximum size must indicated. Then generate new invitations such that:

(A) For each host, determine if committed space is sufficient without displacing other AU's harvested by that host. If a host has sufficient committed space, it increases allocated space to new size, and continues harvesting. No new invitation change needed if $k=4$ hosts are still harvesting.

(B) If a host does not have sufficient committed space, it stops harvesting the AU while retaining old files. New invitations are issued to bring $k=4$ hosts harvesting each AU, without displacing any other current assignments or deleting files.

(C) If $k=4$ hosts cannot be found to harvest an AU, determine the minimum number and size of hosts that would have to be added.

AU2 {lockss-1, lockss-2, props, dris}

Promised: 15
New max size: 19

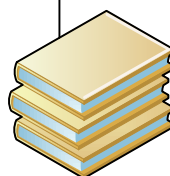
(A)

- lockss-1: Harvesting 210, committed 1700.
Can handle increase.
Increase allocation, harvesting 214, committed 1700. $k=4$.
- lockss-2: Harvesting 210, committed 1700.
Can handle increase.
Increase allocation, harvesting 214, committed 1700. $k=4$.
- props: Harvesting 34, committed 39.
Can handle increase.
Increase allocation, harvesting 38, committed 39. $k=4$.
- dris: Harvesting 36, committed 39.
Cannot handle increase. $k=3$.

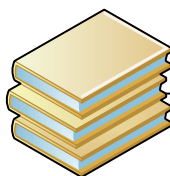
Fewer than $k=4$ hosts are harvesting AU2.

(B)

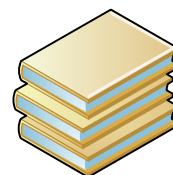
- *lockss-0 is harvesting 75, committed 100.
Invited lockss-0 (harvesting, 94, committed 100). $k=4$.



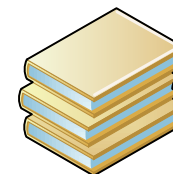
Title	AU0
Size	170



Title	AU1
Size	21



Title	AU2
Size	15
New Max Size	19



Title	AU3
Size	4

Text at <https://wiki.hmdc.harvard.edu/DPP/index.php?pagename=DPP.UseCases>

Data-PASS SSP Use Case 5 AU3

lockss-0.icpsr.umich.edu

IP Add 141.211.192.29
Commitment 100



lockss-1.icpsr.umich.edu

IP Add 141.211.192.52
Commitment 1700



lockss-2.icpsr.umich.edu

IP Add 141.211.192.53
Commitment 1700



props.irss.unc.edu

IP Add 152.2.32.234
Commitment 39



dris.irss.unc.edu

IP Add 152.2.32.233
Commitment 39



haar.irss.unc.edu

IP Add 152.2.32.207
Commitment 200



Machine Ownership Key	
Yellow	ICPSR
Blue	Odum Institute

Invitation	AU3	Invitation	AU0, AU1, AU2, AU3	Invitation	AU0, AU1, AU2, AU3
Newly Invited	AU1, AU2	Newly Invited	0	Newly Invited	0
Deinvited	AU0	Deinvited	0	Deinvited	0

Invitation	AU2, AU3
Newly Invited	0
Deinvited	AU1

Invitation	AU1
Newly Invited	0
Deinvited	AU2

Invitation	AU0
Newly Invited	0
Deinvited	0

SCHEME UC5

Let there be $h=6$ hosts and $n=4$ AU's. Let invitations be previously assigned accepted as follows:

When an AU exceeds its promised size, a new maximum size must indicated. Then generate new invitations such that:

(A) For each host, determine if committed space is sufficient without displacing other AU's harvested by that host. If a host has sufficient committed space, it increases allocated space to new size, and continues harvesting. No new invitation change needed if $k=4$ hosts are still harvesting.

(B) If a host does not have sufficient committed space, it stops harvesting the AU while retaining old files. New invitations are issued to bring $k=4$ hosts harvesting each AU, without displacing any other current assignments or deleting files.

(C) If $k=4$ hosts cannot be found to harvest an AU, determine the minimum number and size of hosts that would have to be added.

AU3 {lockss-0, lockss-1, lockss-2, props}
Promised: 4
New max size: 5

(A)

- lockss-0: Harvesting 94, committed 100. Can handle increase. Increase allocation, harvesting 95, committed 100. $k=4$.
- lockss-1: Harvesting 214, committed 1700. Can handle increase. Increase allocation, harvesting 215, committed 1700. $k=4$.
- lockss-2: Harvesting 214, committed 1700. Can handle increase. Increase allocation, harvesting 215, committed 1700. $k=4$.
- props: Harvesting 38, committed 39. Can handle increase. Increase allocation, harvesting 39, committed 39. $k=4$.

$k=4$ hosts are still harvesting AU3!

Title	AU0
Size	170

Title	AU1
Size	21

Title	AU2
Size	19
New Max Size	

Title	AU3
Size	4
New Max Size	5

Text at <https://wiki.hmdc.harvard.edu/DPP/index.php?pagename=DPP.UseCases>