AIX Live Update and new features and improvements

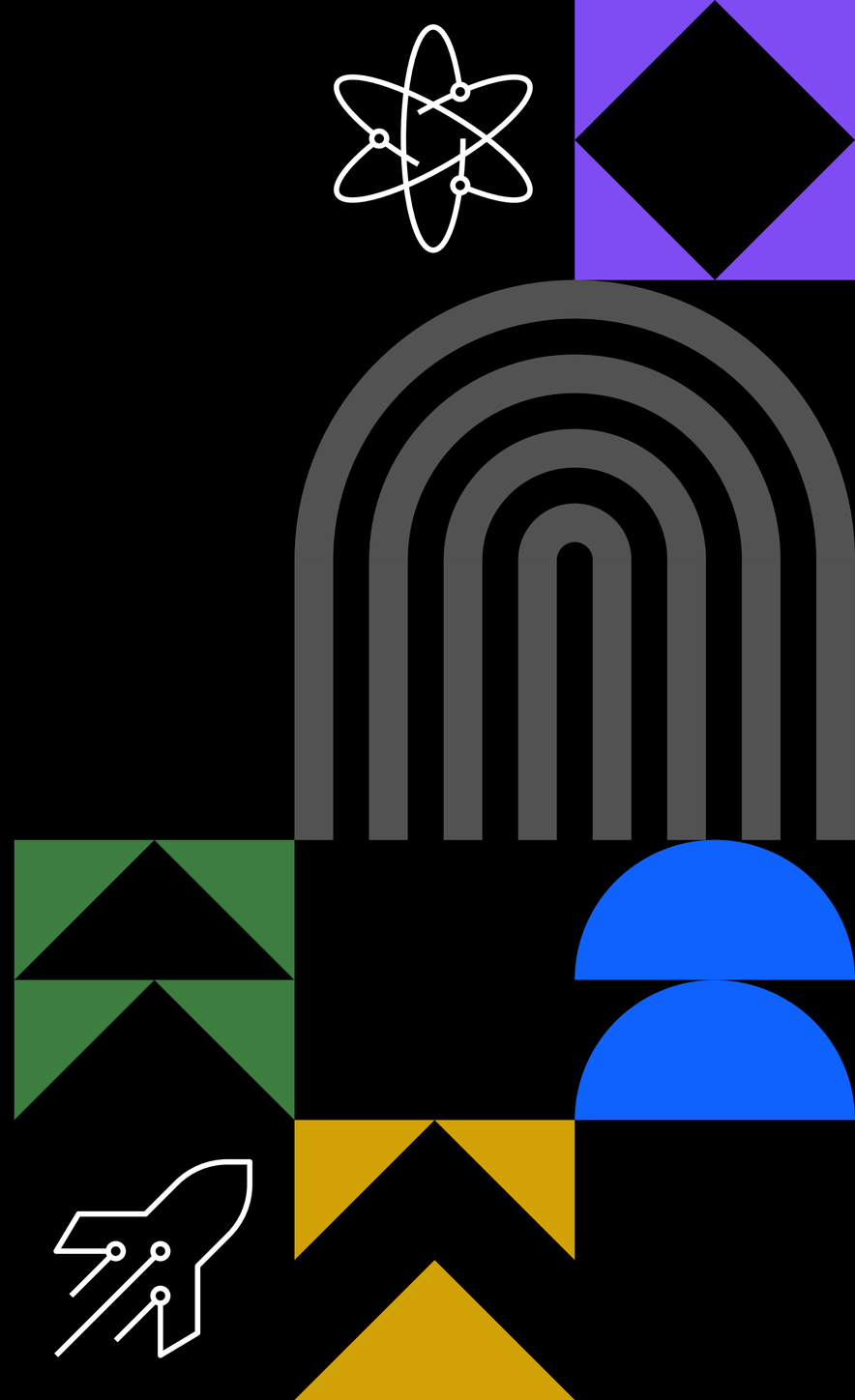
Christian Sonnemans
ASN bank
Senior Unix engineer

Co-Speaker
Chris Gibson
IBM Power/AIX SME
IBM Power Technical Training

Imagine a world without reboots!

Agenda & what you will learn

Agenda



01 Why should we use Live Update

02 Basics Live Update

03 Live Update improvements

04 Live Library Update

05 Ipsec compatible with Live Update

06 Profile settings and Live Update

07 Q&A / continue your learning journey

What you will learn in this session

01 Live Update the basics

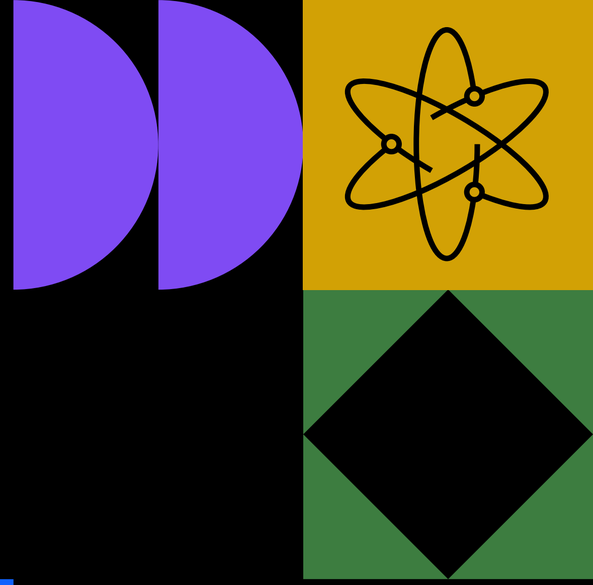
02 Making Live Update robust

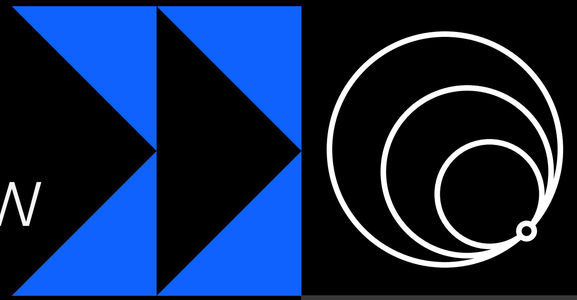
03 Live Update and new technology

04 New technology = faster update window

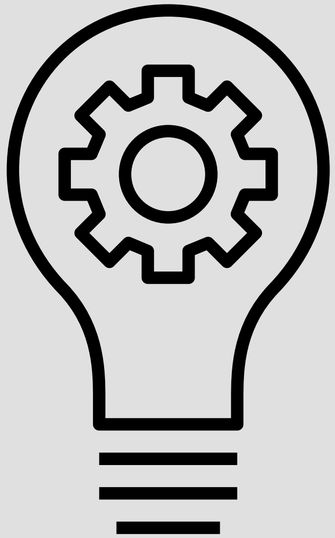
05 Methods to create zero downtime

06 Your feedback to IBM developers is important

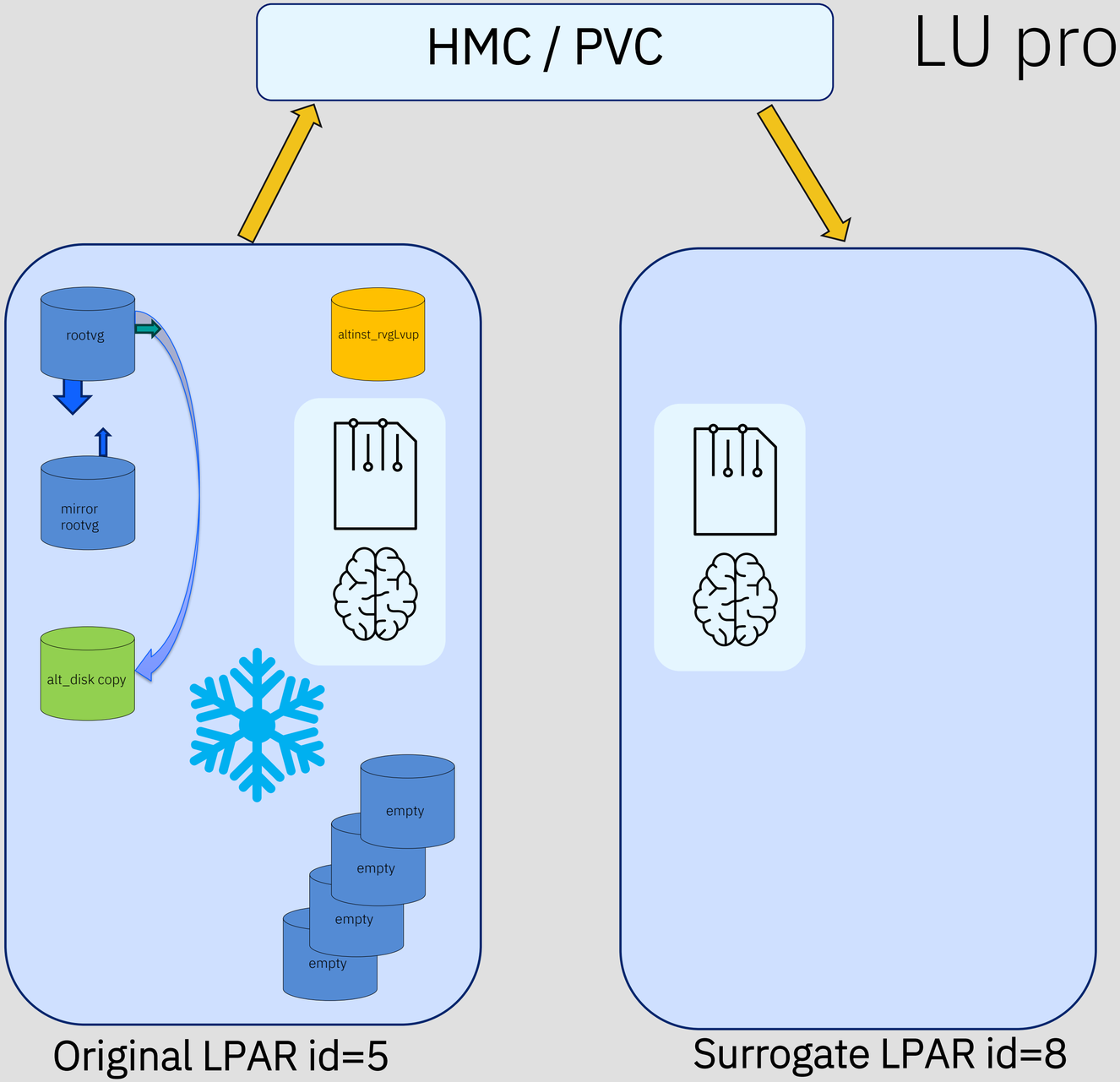




Live Update the basics



LU process steps part1



Legend pictograms

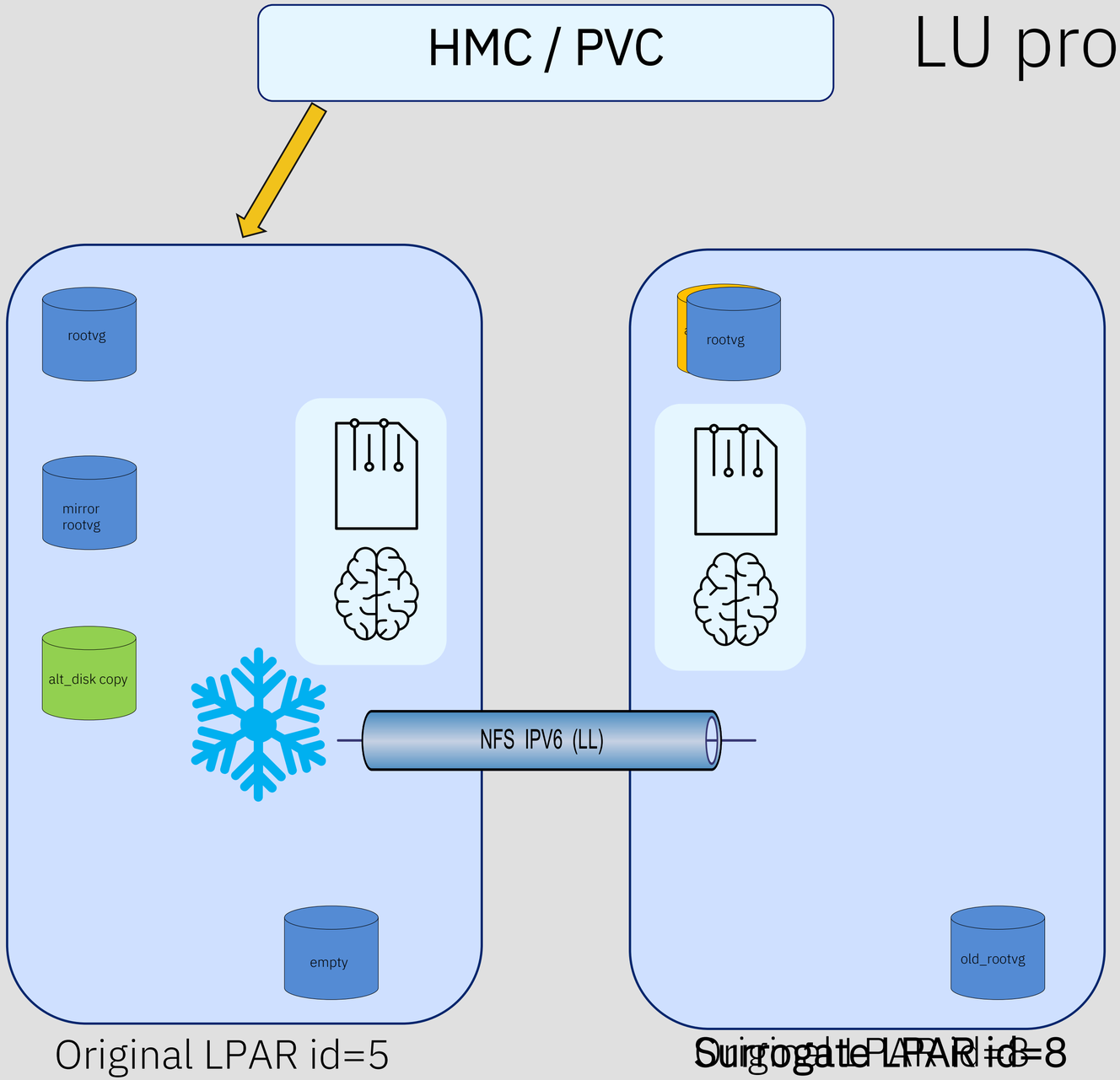
Memory

Process and thread status

Frozen state

Same as blackout

LU process steps part2



Legend pictograms

Memory

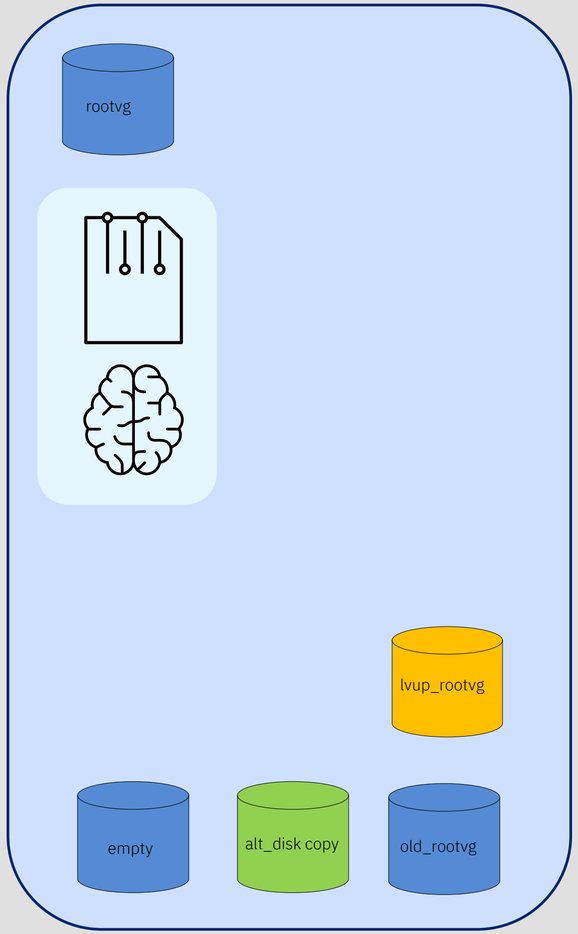
Process and thread status

Frozen state

Same as blackout

HMC / PVC

LU process steps part3



Original LPAR id=8

Legend pictograms

Memory

Process and thread status

Frozen state

Same as blackout

Using NIM or PVC for LU
Using ssh scripts for LU
Using Ansible for LU

Different methods initiate an LKU

Below the preview command for efixes:

```
nim -o cust -a live_update=yes \
-a filesets="$efix_list"\
-a lpp_source=$lpp_source \
-a installp_flags="-p" \
-a live_update_data=$nim_res $lpar
```

Without the -p flag for actual update

Below the preview command for lpp fixes:

```
nim -o cust -a lpp_source=72000402_lpp_source \
-a installp_flags="-p" \
-a fixes=update_all \
-a live_update=yes \
-a live_update_data=liveupdate_data_<LPAR-name> <LPAR-name>
```

Without the -p flag for actual update

Setting up complete NIM environment for LU see my blog:
https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2021/03/27/live-update-cookbook

Below the commands for a manually start of a LKU

```
hmcauth -u liveupdate -p <passwd> -a <ip_address_hmc>
hmcauth -l

Or use pvcauth if you like to use PowerVC instead of HMC.

Update with update_all -d <path>

geninstall -p -k    ← Preview
geninstall -k       ← initiate the LKU
```

Or in remote shell script:

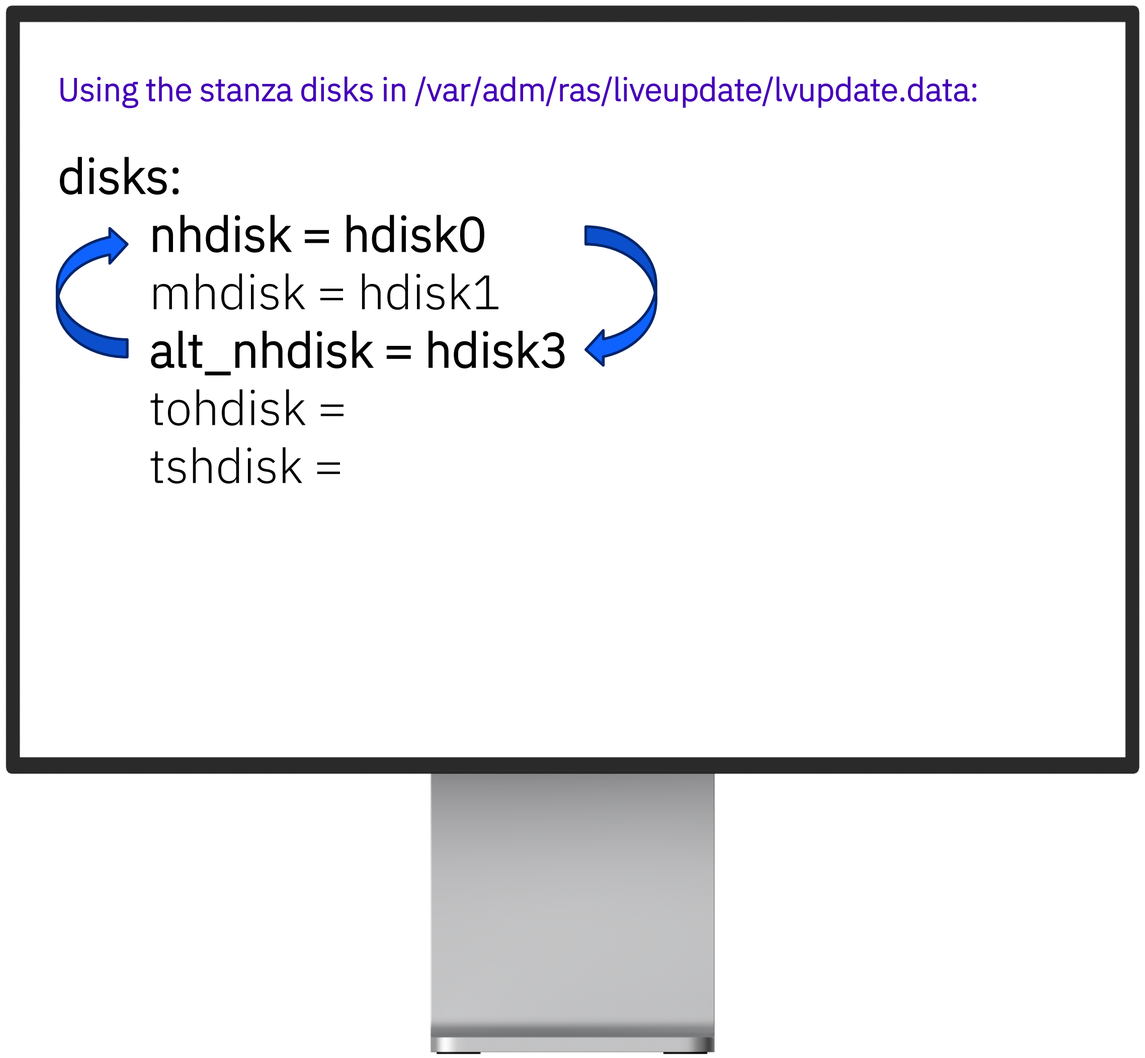
```
ssh -q $lpar geninstall -p -k    ← Preview
ssh -q $lpar geninstall -k       ← initiate the LKU
```

Below the link for ansible by Chris Gibson
https://community.ibm.com/community/user/blogs/chris-gibson1/2024/08/09/performing-aix-live-update-with-ansible-automation

Automatically switch disks with consecutive LKU’s

Using the stanza disks in /var/adm/ras/liveupdate/lvupdate.data:

```
disks:
        nhdisk = hdisk0
        mhdisk = hdisk1
        alt_nhdisk = hdisk3
        tohdisk =
        tshdisk =
```



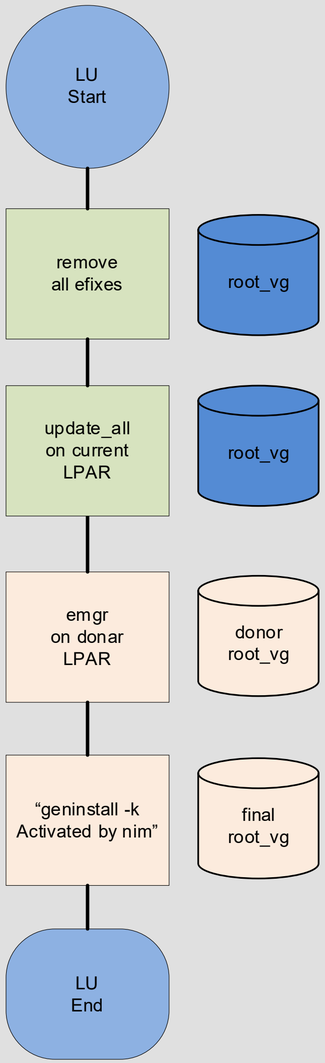
This method makes it possible switch automatically between **hdisk0** and **hdisk3**.

With three empty disk, you do no longer have to edit this stanza every time.

The **lvup_rootvg** (disk that stays occupied)
Flips between **hdisk0 and hdisk3 (nhdisk & alt_nhdisk)**
The rootvg stays the same **(mhdisk)**

Note: the **lvupdate.data** file can also defined as **NIM resource**

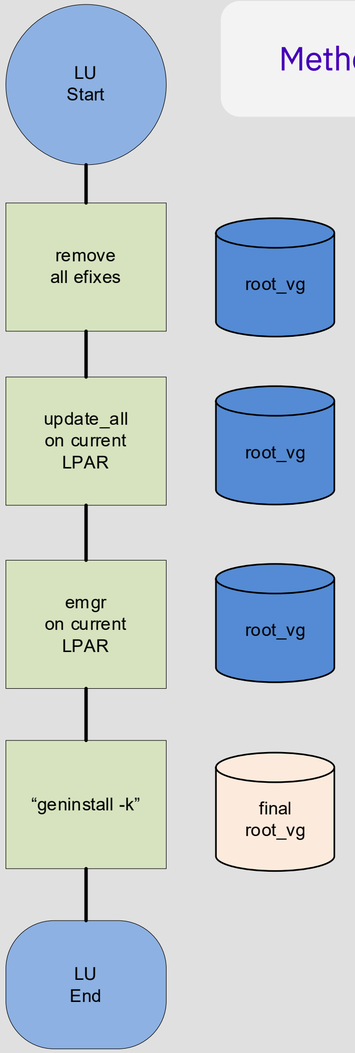
How to apply **LPP, SP, TL** patches and **efixes** in one run?



Method for NIM

Method one

1. Remove all efixes first from current LPAR
2. Run **update_all** from a NIM resource on the current LPAR
3. Apply efixes on donor (surrogate) LPAR
4. Run a LU via NIM that will apply lpp and efix packages

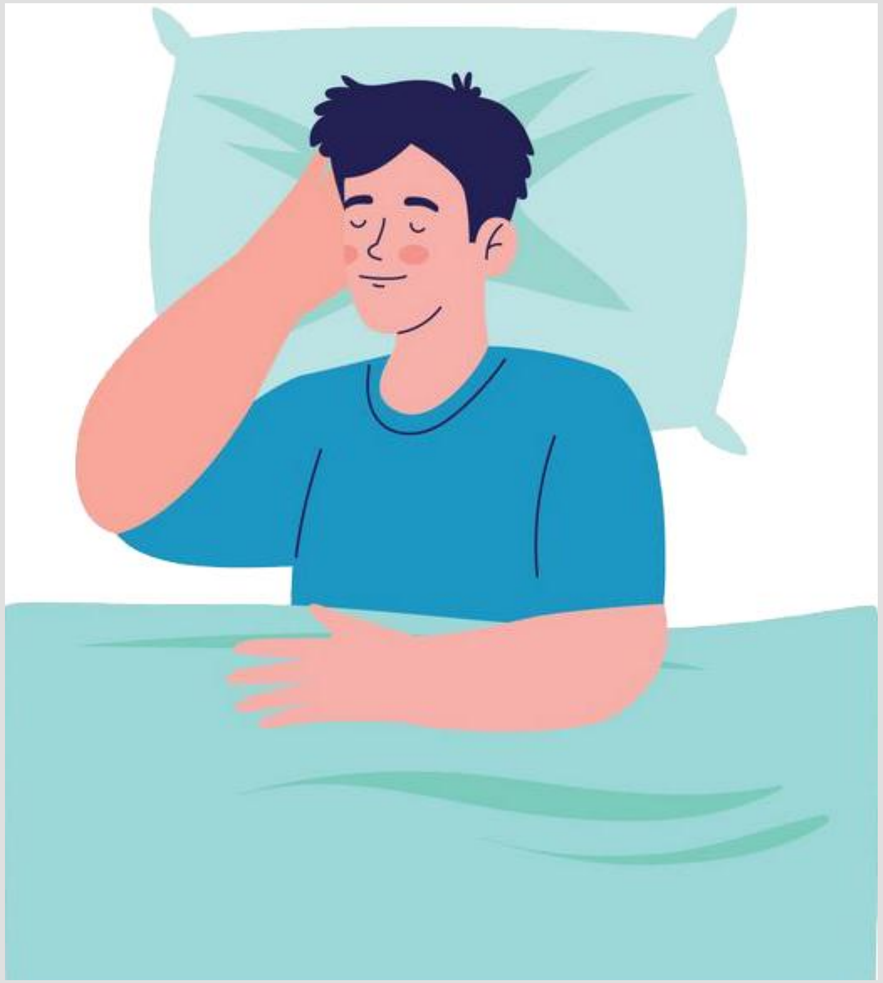


Methods using manual, remote ssh, ansible, or PowerVC

Method two all on original

1. Remove all efixes first from current LPAR
2. Run **update_all** from a NIM resource on the current LPAR
3. Apply efixes running LPAR
4. Run a LU via script that will apply lpp and efix packages

Why I was so eager to get LU to work?

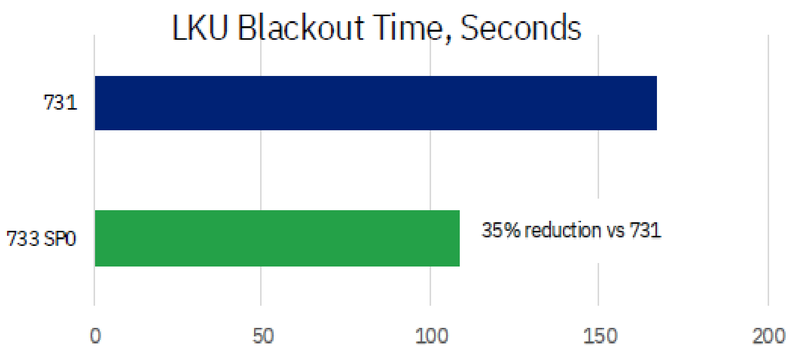


AIX 7.3 improvements = reduced ❄ ”blackout” window

Note: AIX 7.3 SP0 compared to SP1

On my small test LPAR: shared mode=Uncapped smt=8 lcpu=8 mem=5120MB psize=1 ent=0.20

Large LPAR test (source IBM)



LPAR details: Power9, 10 cores, SMT8, 300 GB RAM, 410 GB rootvg, 256 GB dump, 181 LUNs, 29 VGs, 107 JFS2 mounts, 5 active Oracle instances, 2 OAST workload instances
AIX 733, 733 POC1, and 733 SP1 tests use a pre-release builds. 733 SP1 test is with Power10.

My tests on small LPAR with *SP0*

7300-02-02-2420:
Average Blackout=18.7 Sec.

7300-03-00-2446
Average blackout=18.2 sec

Compared with AIX 7.2 TL05 SP9 12.3 sec

This slight increase had to do with a lot of optimalisations such as shared memory dump optimization +3 sec

My tests on small LPAR with *SP1/SP2*

7300-03-01-2520:

IBM LDAP / DB2 servers
Average backout=11.5 sec

Overall time 11 minutes.
Compared to 7.2 avg 22 min.

The four stages to make LU predictable and robust

Stage one
Pre check stage:
Have we met all the prerequisites?
Do we have the right resources?
Is the patient awake but not stressed
In this stage nothing is changed or stopped

Stage two
Prep stage:
Stop necessary processes, devices, but not the vital functions for the application.

In this stage monitoring tools are stopped, ~~ip-sec,~~ TE, so that we can run a LU preview

Stage three
The patch / LU stage:
Do the actual surgery.
Take over vital functions.

In this stage the actual LU takes place with a short freeze

Stage four
Post stage:
After the surgery normal life resumes for the patient.
Start and check all other functions and do aftercare.
In this stage the monitoring starts, TE, ~~ip-sec~~, etc.

But what if...

Something does go wrong?

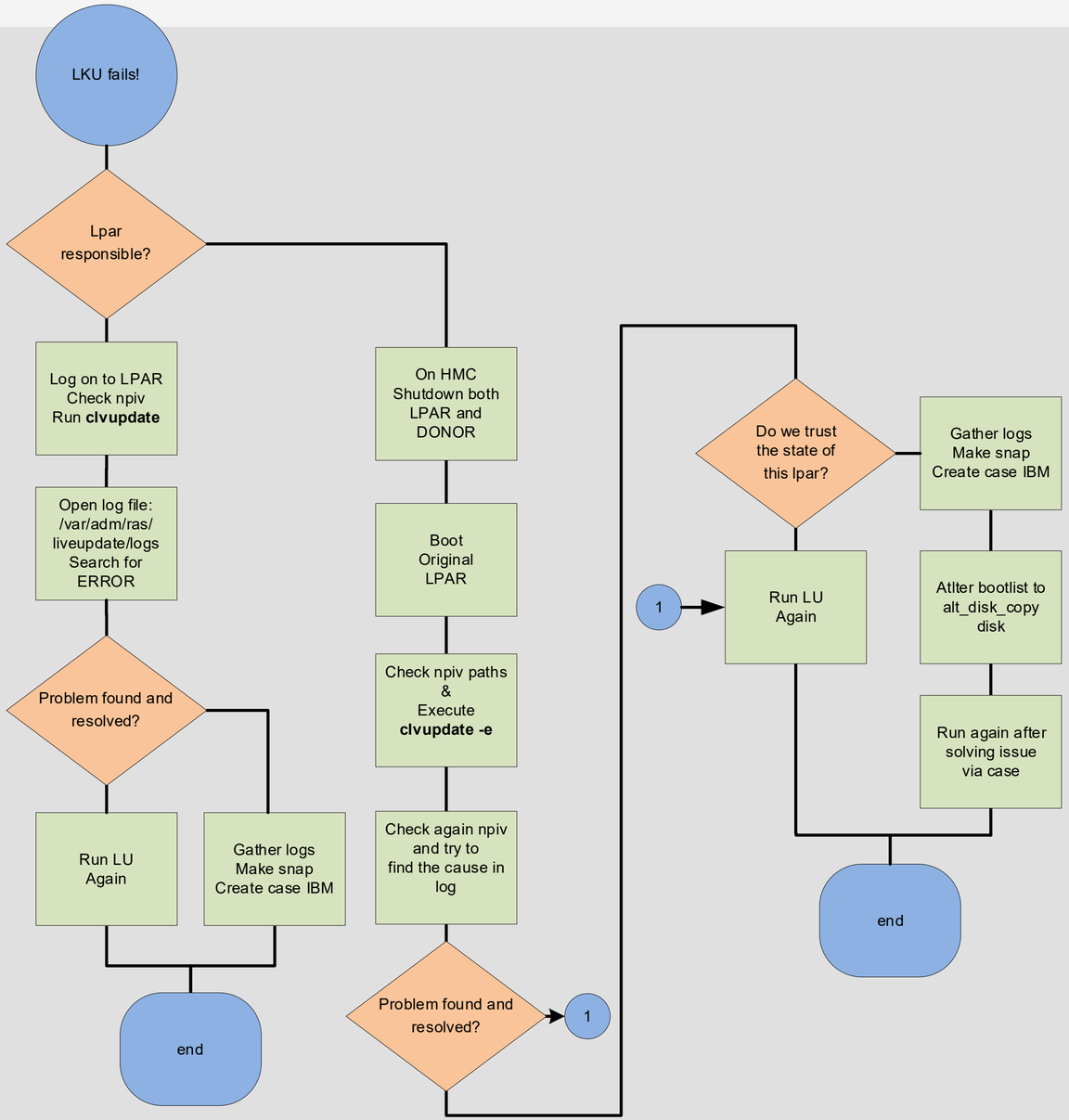
```
1020-089 Internal error detected. [02.335.0458]
........
1430-030 An error occurred while moving the workload.
1430-045 The live update operation failed.

Initiating log capture.

Local log capture in progress and can be tracked from:
        /tmp/snapimg/lu_2024-08-05_10-08-
15_capture.log
On completion, log archive will be stored as:
        /tmp/snapimg/lu_2024-08-05_10-08-15_snap.pax.Z
```

But what if...

Something does go wrong?



But what if…

Something does go wrong?

Step 1: verify NPIV paths

- Verify NPIV paths (**`lspath`** / **`lsmpio`**)
- If paths not ok run **clvupdate** and **clvupdate -e** (after a reboot)

Step 2: Check the local log file(s)

- view **/var/adm/ras/liveupdate/logs/lvupdlog** (search for **ERROR**)
- check **`errpt`**

Step 3: Problem found in log ?

- Check background logging LU is still running → **`ps -ef | grep local_capture`**
- Check background snap is still running → **`ps -ef | grep snap`**
- Wait until log captures ended.
- Resolve the problem and run LU again. If not resolvable go to step 4.

Some AIX 7.3 >TL03 Quirks

FC vio setting for some specific adapters

With LKU during post phase very occasionally on npiv (mpio) path lost.

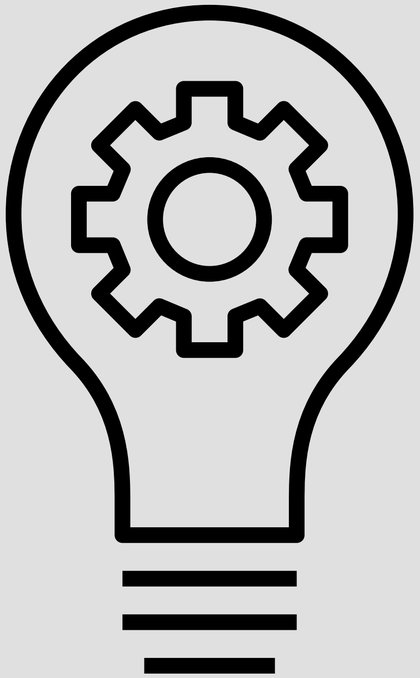
Cause: one vfchost error during the LKU Cleanup phase Post phase of LKU.
Solution: set following setting on vios: **chdev -dev fcs# -perm -attr sw_prli_rjt=yes**
Warning !Only for **some specific VIOS FC adapters**
For more detailed information: https://community.ibm.com/community/user/blogs/christian-sonnemans1/2026/02/05/vio-fc-quirks-of-lku-and-aix-73-tl0304 and Error LABEL: VIOS_VFC_CLIENT_FAIL IDENTIFIER: D119376C logged on VIO server

Work in progress

race condition between concurrent LKU and validation processes: (looks similar then above but is a different error!)

During LKU cleanup phase, resulting in missing fscsi paths for the affected virtual Fibre Channel adapter on the NPIV LPAR.
workaround: Run the cfgmgr command on the affected NPIV client LPAR to restore the missing fscsi paths. Apar: IJ58268: VIOS_VFC_CLIENT_FAIL WITH NPIV_ERR_0185
More detailed info: IJ58268: VIOS_VFC_CLIENT_FAIL WITH NPIV_ERR_0185 LOGGED DURING LKU

New AIX 7.3 features



Live Library Update

PLEASE NOTE: This feature is supported and official released in TL04 5-Dec-2025.

But the only real tests we can run when AIX 7.3 TL04 SP1 is available !

What is LLU Live Library Update ?

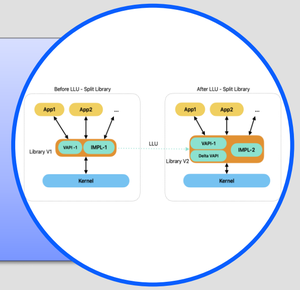
AIX

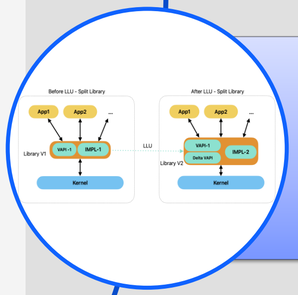
Live Library Updates is a new technology

AIX 7.3 TL04 SP1 and higher

Run independent of Live Update

(LU & LLU are separate things!)

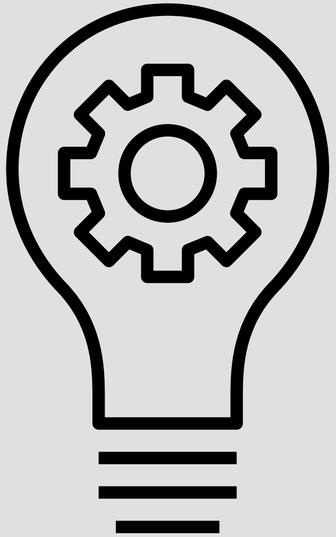


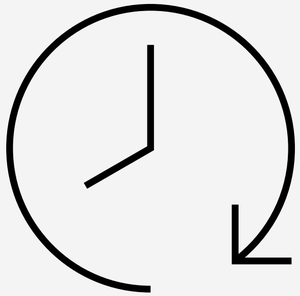


replaces live shared libraries while processes are running and using shared libraries

LLU can be run after a LU they are a good marriage!

Live Library Update - The basics

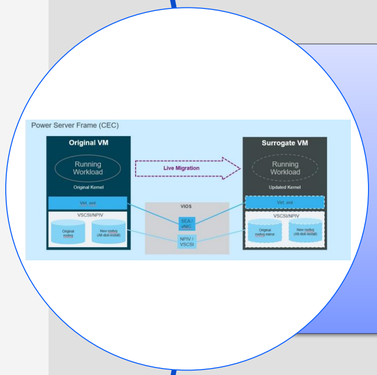




When? Could or should we use it?

AIX

After applying new software SP, TL, or efixes HIPER fixes than also updates shared libraries

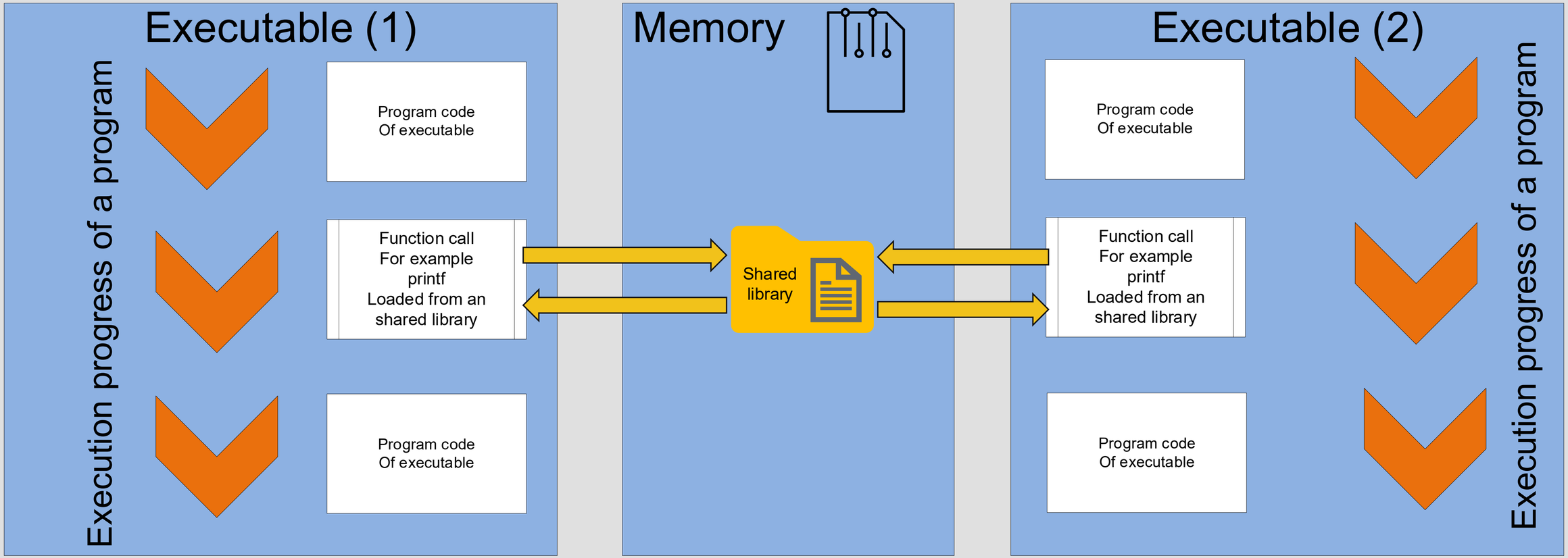


After a LU is being used, for example for of one of the above updates or patches

PowerVM

On non-virtualized LPAR’s after upgrading software (see first item)

Shared libraries explained



What are shared libraries anyway? A quick recap!

New feature LLU = requires a new AIX Loader

LLU capability of a process is decided when a process is loaded.

<u>Preconditions:</u>

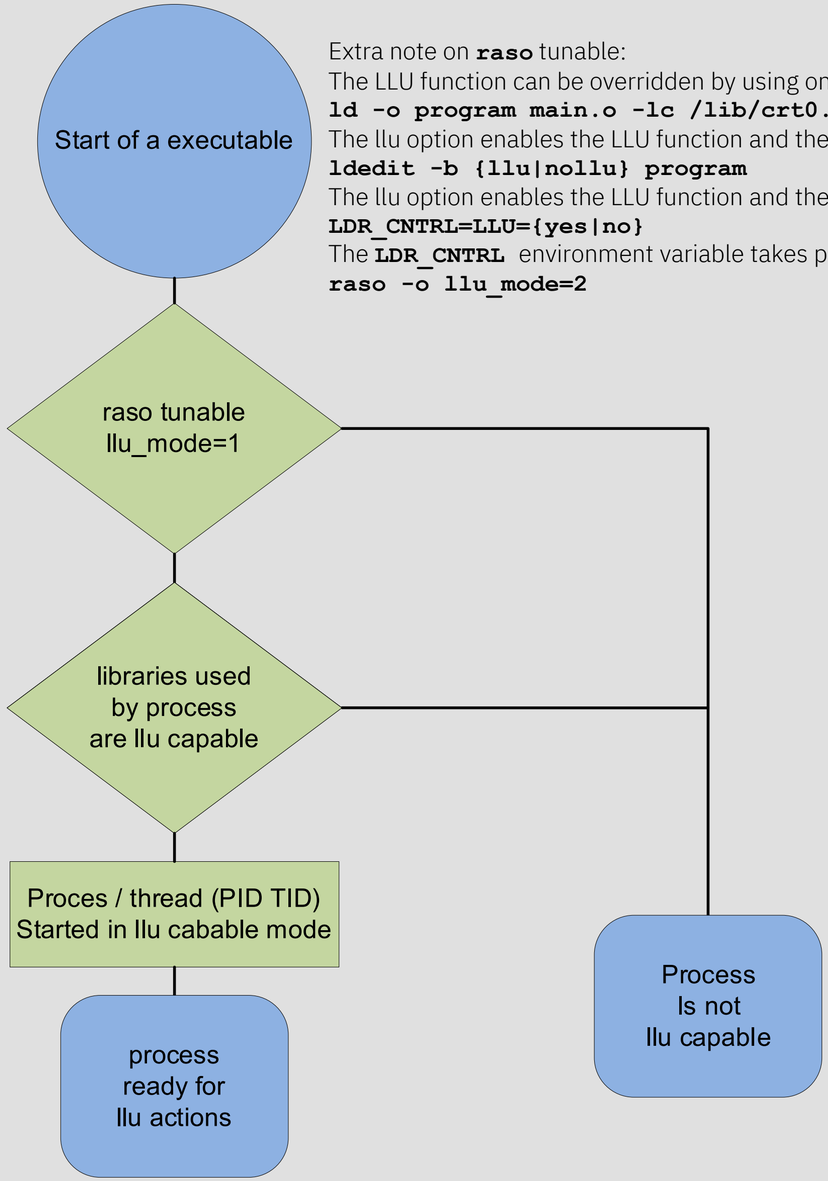
The system wide **raso** tunable **llu_mode** should be set to **1**

raso -p -o llu_mode=1

Pitfall: only **new started processes** are LLU capable after this command!! In other words, **one reboot** is necessary.

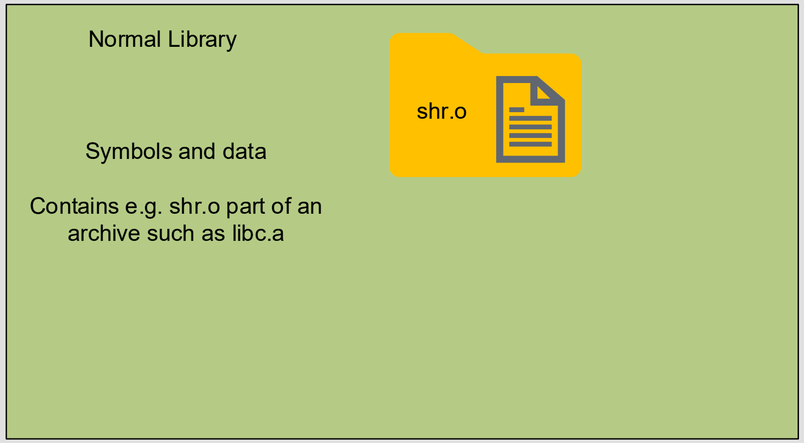
The libraries used by process should be LLU capable. Currently **libc.a** and **libpthreads.a** are the two libraries that are LLU capable.

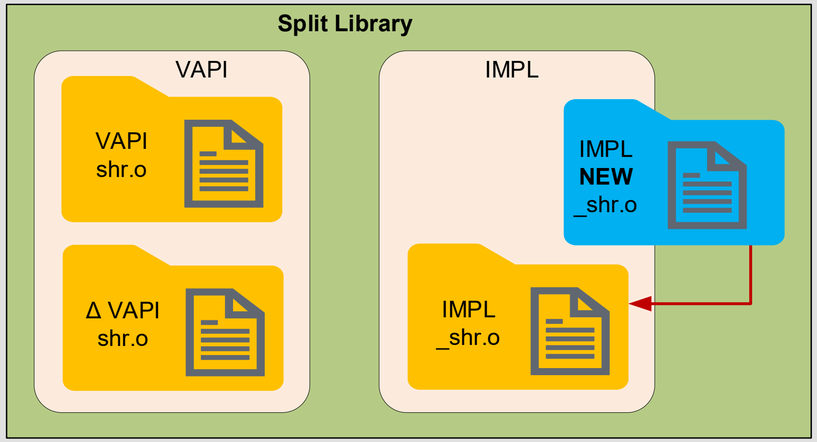
Note:
From AIX 7.3 TL03 **SP1**
Default **llu_mode=2** in AIX 7.3 TL03 SP1



Extra note on **raso** tunable:
The LLU function can be overridden by using one of the following commands or methods:
ld -o program main.o -lc /lib/crt0.o -b {llu|nollu}
The llu option enables the LLU function and the nollu option disables the LLU function.
ldedit -b {llu|nollu} program
The llu option enables the LLU function and the nollu option disables the LLU function.
LDR_CNTRL=LLU={yes|no}
The **LDR_CNTRL** environment variable takes precedence over the LLU program attribute.
raso -o llu_mode=2

LLU requires libraries in split mode





What are split libraries?

```
# ar -tv libc.a | grep shr.o
r-xr-xr-x   2/2    1228445 Oct 14 14:05 2024 shr.o
r-xr-xr-x   2/2    6249189 Oct 14 14:05 2024 _shr.o
```

Preserved across LLU operation

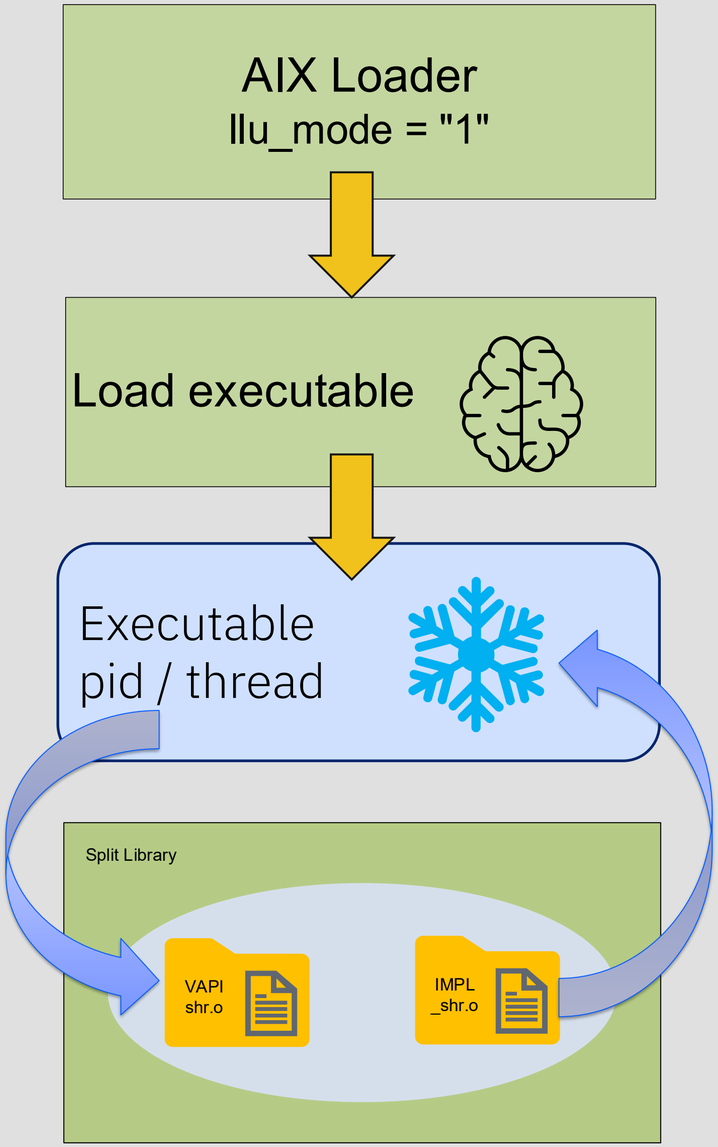
VAPI = Virtual Application Layer Contains exported symbols and data that should **preserved during LLU.**
The **shr.o** part of the archive **shr.o** is the **VAPI object (VAPI)**

Additional Δ VAPI object symbols (**if any**) are added through delta VAPI object.

Modified across LLU operation

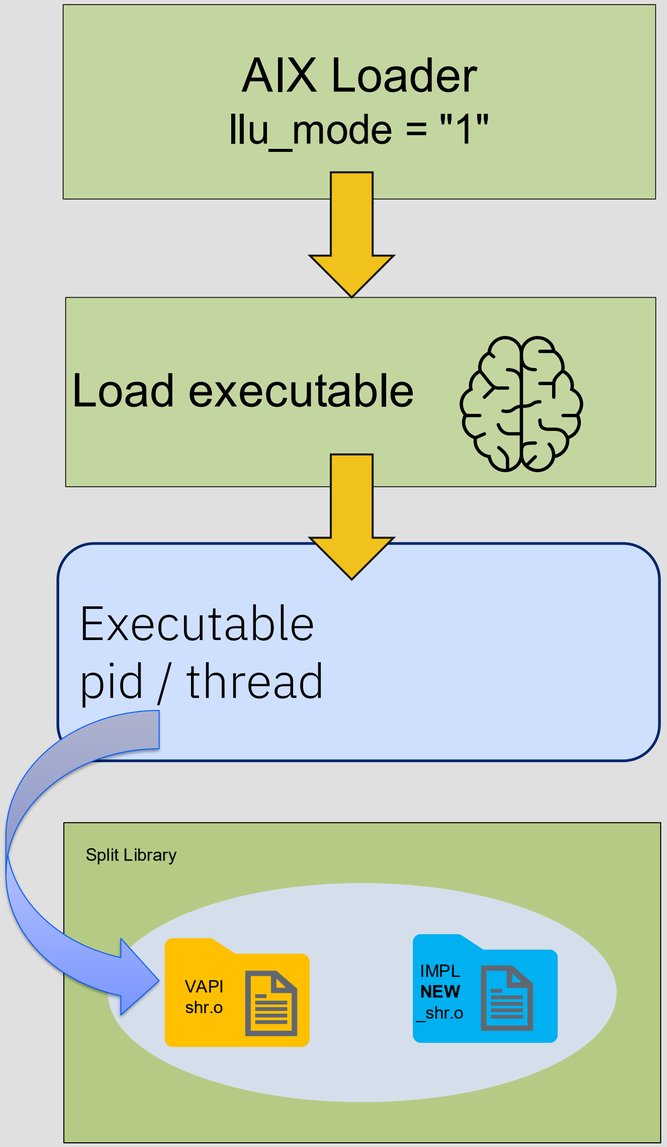
Implementation = contains the library implementation code

The **_shr.o** which is part of the same **libc.a** (archive) is called the **implementation object. (IMPL)** object is replaced with new code after the LLU operation.



LLU process steps part1

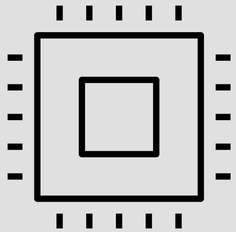
Process is started with AIX loader LLU capable llu_mode=“1”



LLU process steps part2

LLU can be run after an LU they are a good marriage!

But why should (could) we use LLU after a LU operation?



Use case LLU running after an LU operation

- We use Live Update to update to a new SP or TL or to apply kernel fixes.
- During this software is updated read executables and libraries.
- After a LU processes are “resumed”
- Remember the slide with post actions, some processes will be still attached to the old libraries (**genld -u**)

- Before 7.3 TL3 we used a postscript to scan those scripts and restart most of those processes.

sshd is an example.

REMEMBER LLU running on TL04 SP00
only two LLU-capable libraries in 7.3 TL04 SP0:
libc.a and libpthreads.a.

- But more and more split libraries will be come available in the future (not in TL04).

How to run Live Library updates automatic after a Live Update?

On the LPAR

1. **`raso -po llu_mode=1`**
- Note the “-**p**” flag updates the **/etc/tunables/nextboot** and **nextliveupdate** files so that it survives a Live Update

2. Edit the **/var/adm/ras/liveupdate/lvupdate.data file** and add the following stanza:

```
llvupdate:
        llu = yes
        retries = 3
        timeout = 30
```

A Live Library Update will be performed immediately after the Live Update operation

Logs for the LLU can be found in:

/var/adm/ras/liveupdate/logs

How to run Live Library updates manually?

On the LPAR

1. **`raso -po llu_mode=1`**
   - Note the "-**p**" flag updates the **/etc/tunables/nextboot** and **nextliveupdate** files so that it survives a Live Update

2. Run the commands:
   `llvupdate -P` → preview
   `llvupdate -a -n2 -t60`

This command does a complete scan and tries twice, with a timeout of 60 seconds

3. Check the log:

`cat /var/adm/ras/liveupdate/logs/llvupdlog`

Trace file:

`/var/adm/ras/liveupdate/trace_llu/trcfileLLU`

Important to know!

Live Library Update can also be used on **NON-virtual** LPAR's

Output of a manual run after a SP0 to SP1 update

```
# llvupdate -a -n2 -t30
Non-interruptable live library update operation begins in 10 seconds.
llvupdate preview
Validating new module /usr/lib/libpthreads.a(_shr_xpg5.o)
Validating new module /usr/lib/libc.a(_shr.o)
Adding the process init pid 1 to the list
Validating new module /usr/lib/libpthreads.a(_shr_xpg5.o)
New module has additional exports /usr/lib/libpthreads.a(_shr_xpg5.o)
Validating new module /usr/lib/libc.a(_shr.o)
New module has additional exports /usr/lib/libc.a(_shr.o)
Adding the process secldapclntd pid 3080698 to the list
Adding the process nimon pid 3932504 to the list
Adding the process nimsh pid 4194708 to the list
Validating new module /usr/lib/libpthreads.a(_shr_xpg5_64.o)
New module has additional exports /usr/lib/libpthreads.a(_shr_xpg5_64.o)
Validating new module /usr/lib/libc.a(_shr_64.o)
New module has additional exports /usr/lib/libc.a(_shr_64.o)
Adding the process backup_agent_main pid 4325786 to the list
Adding the process ntpd4 pid 4456872 to the list
Adding the process bootstrap_agent_main pid 5046580 to the list
Validating new module /usr/lib/libpthreads.a(_shr.o)
New module has additional exports /usr/lib/libpthreads.a(_shr.o)
Adding the process portmap pid 5308900 to the list
Adding the process iked pid 6160782 to the list
Adding the process biod pid 6554070 to the list
Adding the process aso pid 6619634 to the list
Adding the process srcmstr pid 6881760 to the list
Adding the process netcd pid 7143692 to the list
Adding the process sshd pid 7602450 to the list
Adding the process ksh93 pid 7864828 to the list
Adding the process BESClient pid 8061234 to the list
Adding the process cpsd pid 13107690 to the list
Adding the process syslog-ng pid 14746054 to the list
------------------------------------------------------------
************************* TRY 1 *************************
======== NOTIFY PHASE BEGIN ======== | try 1 | tonotify:18
======== NOTIFY PHASE ENDED ======== | try 1 | suspended:18
```

But what if...

Something does go wrong?

possible failures...

LLU might fail in following scenarios:

1. Process is busy in system call or in kernel at the time of LLU
2. Process is continuously executing in implementation layer of library. In the above failure scenarios LLU can be retried using "**-n**" option as shown below:

```
llvupdate -p <pid> -n <reties>
```

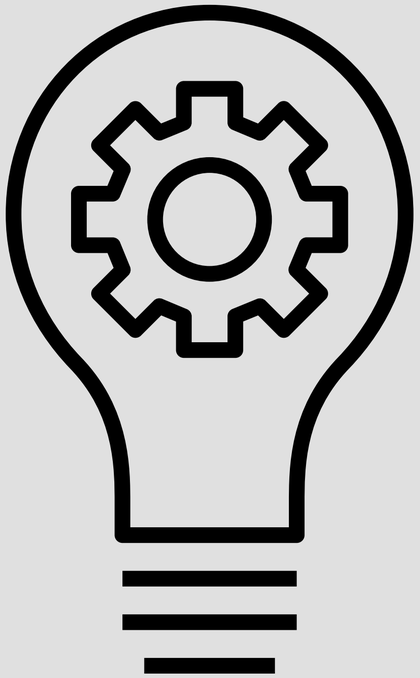
3. In certain rare scenarios of LLU failure, where application processes are left in uncleaned state.

   cllvupdate command can be used to recover and clean the process state before attempting further LLUs.

   Here is the usage of **cllvupdate** command:

```
cllvupdate -u
```

AIX IPSEC is now compatible with AIX LU



What are the benefits for IPSEC IP filters during LU ?

Below AIX 7.3 TL03

Remember four stages prep and post stage:

1. Prep stage: unload all ip filters for example: **`rmfilt -v4 all`**
2. remove ip_sec devices: **`rmdev -dl ipsec_v4`** and **`ipsec_v6`** (if you use ipV6)
3. Run the Live Update
4. Post stage: re-create the ipsec devices: **`/usr/sbin/mkdev -c ipsec -t 4`** (and or) **`-t6`**

Load the ip-fil filter rules again:

dns example filter rule

```
/usr/sbin/genfilt \
-v 4 -a P -s 10.210.252.136 -m 255.255.255.255 \
-d 0.0.0.0 -M 0.0.0.0 -g N -c tcp -p 123 -o any \
-O eq -P 123 -r L -w I -i en5 -D ntp_hostname -n 88
```

AIX 7.3 TL03SP0

No longer need for prep and post stage
Load the ip-fil filter rules again:

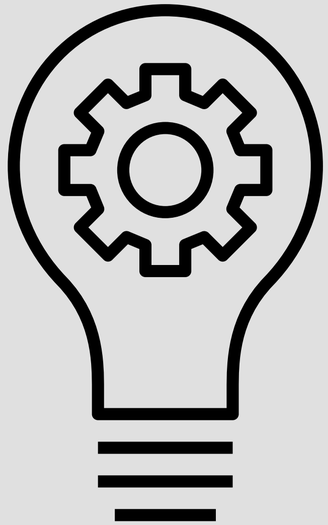
1. For SP0 apply efix: **IJ55348s0a.250804.epkg.Z**
2. Before LU edit the **lvupdate.data** file and update the HMC stanza field

```
general:
        kext_check = yes
        ipsec_auto_migrate = yes
```

3. Just run the Live Update, without stopping the ip-filters and the removal of the ip_sec devices.
4. Special note for 7.3 TL04 ODM change See one of my last blog (default filters do not work)

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2026/05/29/ipsec-ip-filter-issue-on-73-tl04-sp00

Dynamically modifying system resources using LKU with HMC profile



Did you ever run into the limits of your LPAR profile?

For example, max virtual CPU or max memory settings.

Before there was only one solution:

You had to shutdown your LPAR completely!

Set profiles settings and boot again with the new profile.

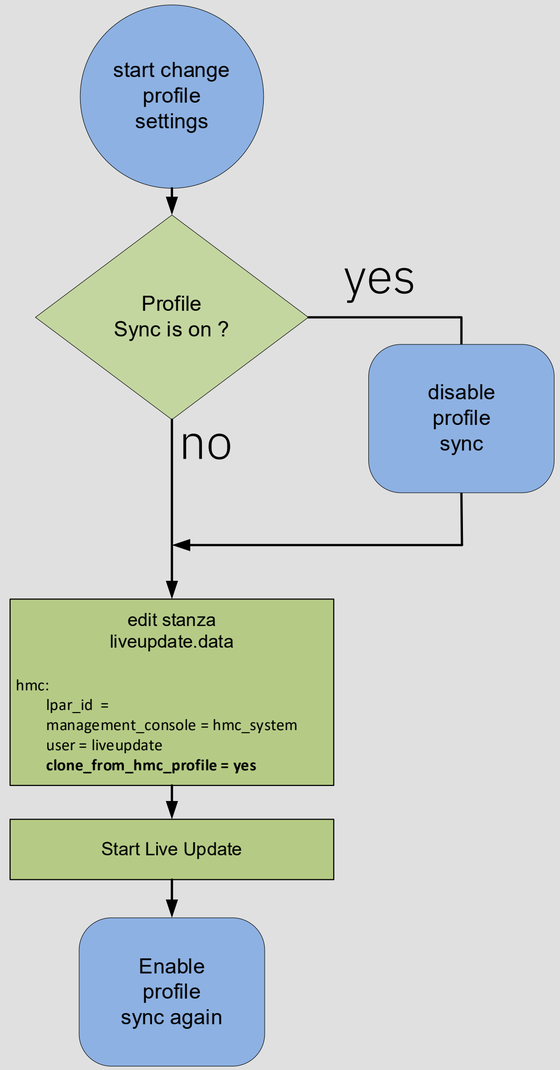
very annoying, shutting down your application or database server for just applying a new profile.

With AIX 7.3 TL3 SP0 and higher.

You can now use Live Update to modify your profile settings. You can now use LU to update it without downtime.

Note: only increasing values are possible!

Workflow: adjusting profile settings with LU



Some limitations with dynamic modifying profiles

- Can only be used to increase values.
- Changing the processor mode for an LPAR from dedicated to shared or vice-versa is not supported or allowed!

Memory configuration settings:

Minimum memory
Maximum memory

Dedicated CPU settings:

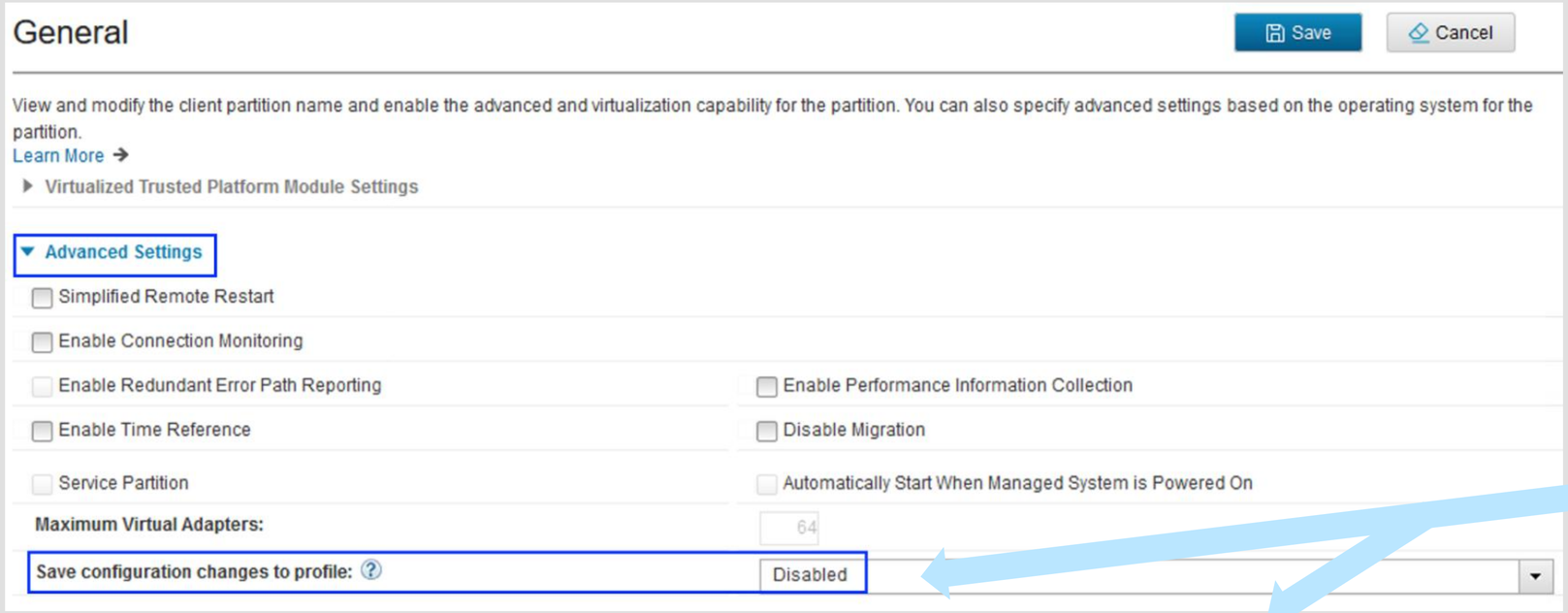
Minimum dedicated processors
Maximum dedicated processors

Shared processor configuration settings:

Minimum shared processing units
Maximum shared processing units
Minimum virtual processors
Maximum virtual processors

Disable and enable profile sync on HMC

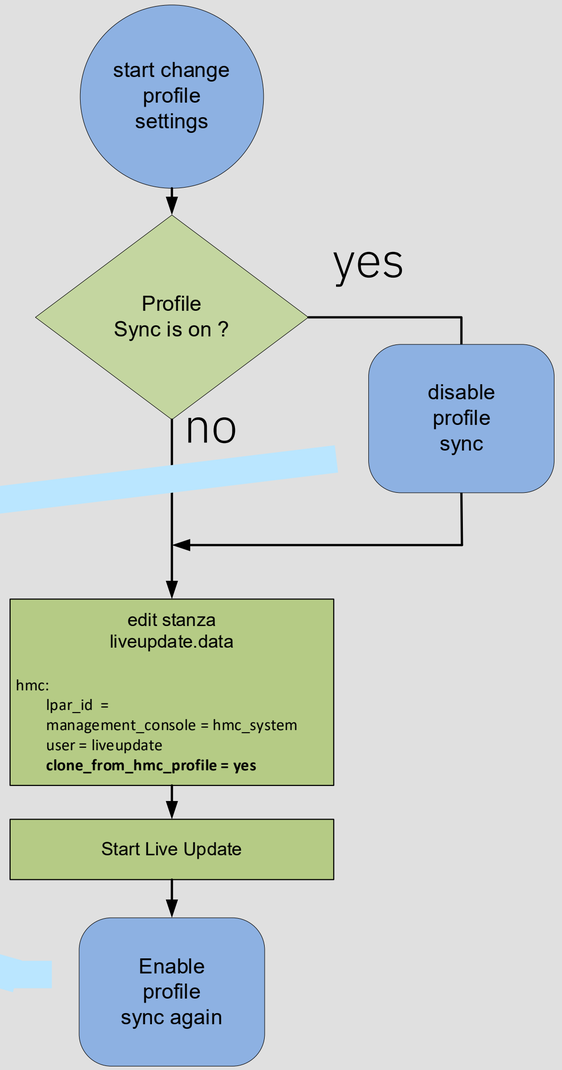
Workflow: adjusting profile settings with LU



chsyscfg -r lpar -m <mgmt-sys> -i "name=lpar_name,sync_curr_profile=0"

Or via cli

chsyscfg -r lpar -m <mgmt-sys> -i "name=lpar_name,sync_curr_profile=1"



and now glue everything together: **lvupdate.data**

```
general:
      kext_check = yes
      cpu_reduction = no
      ipsec_auto_migrate = yes

hmc:
      lpar_id  =
      management_console = hmcname
      user = lvupdat_user
      clone_from_hmc_profile = yes

llvupdate:
      llu = yes
      retries = 4
      timeout = 30

# trace:
#      trc_option = -anl -T 20M -L 40M

disks:
   nhdisk = hdisk1
   mhdisk = hdisk0
   alt_nhdisk = hdisk2

#To move for pagining in non rootvg
#    tohdisk =
#    tshdisk =
```

Enable auto migrate IP-SEC setting without stopping ip-filt

Or ip-tunneling

Update profile setting during a LKU (max limits)

Stanza for Live Library Updates

Enable this stanza for tracing

Nice trick for rotating the temporary surrogate boot disk when using consecutives LKU's

If you have paging space on a non rootvg you have to use this stanza, otherwise you don't have to use it.

Q&A

What I ask from you is, are you inspired and ready to start with LKU?
We are getting inspired by your questions, please ask!
We love to Xchange Tech.
Send me a message via linked-in or email if you would like to stay in contact.
Of course, I am always in for a nice tech talk face to face so do not hesitate to ask!!

Christian Sonnemans

ASN bank

Senior Unix engineer

christian.sonnemans@asnbank.nl

Chris Gibson
IBM Power/AIX SME
IBM Power Technical Training

LinkedIn
Christian Sonnemans
senior beheerder | ontwerper IBM Power Systems | Unix AIX / Linux

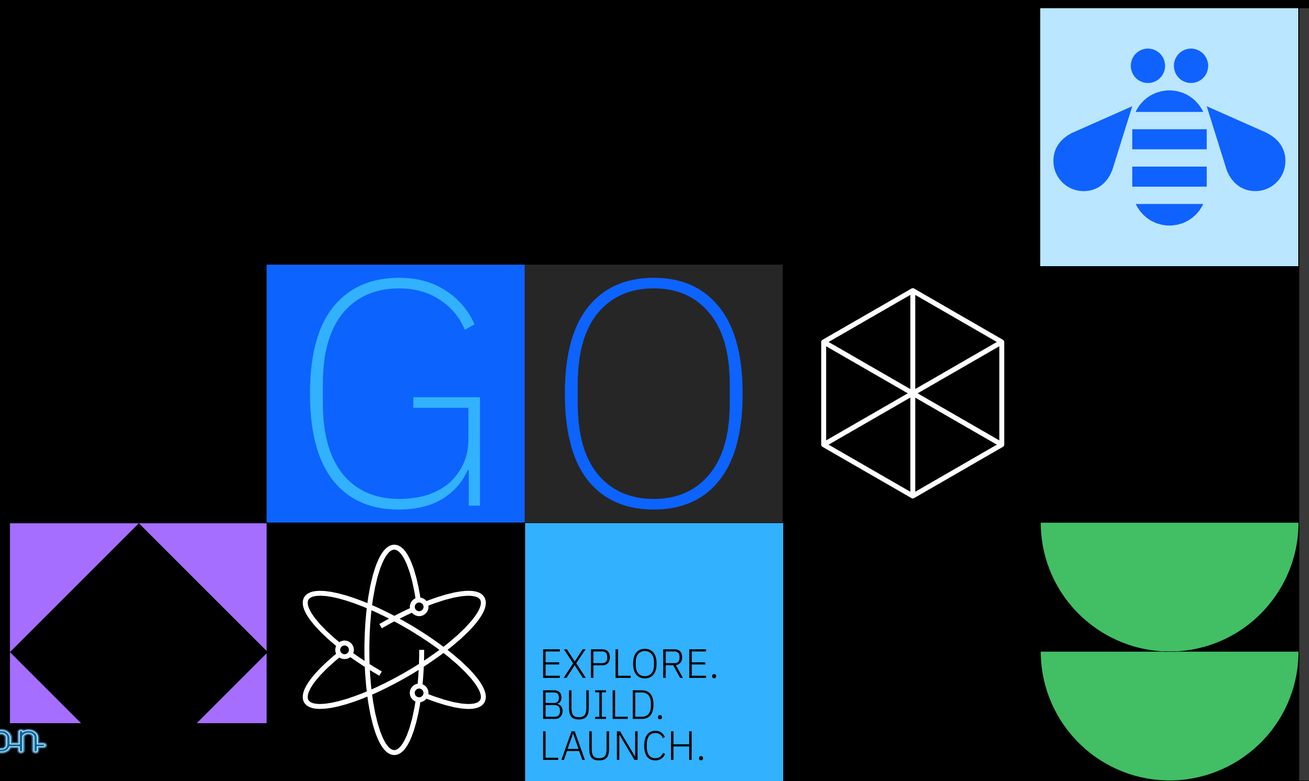
GO
EXPLORE.
BUILD.
LAUNCH.

Please provide your feedback!
Tell me (us) if you liked ☺ it or not ☹

Thank you

Christian Sonnemans
ASN bank
Senior Unix engineer
christian.sonnemans@asnbank.nl

Chris Gibson
IBM Power/AIX SME
IBM Power Technical Training



blogs to read:

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2025/01/30/testing-aix-73-tl03-new-features-and-lku

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2025/02/21/testing-aix-73-tl03-new-features-and-lku-part2

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2025/03/20/ip-sec-ip-filtering-on-vios-41

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2025/04/18/patching-aix-and-vios-fixes-the-right-way

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2025/05/19/ideas-to-create-a-quantum-safe-aix

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2025/06/09/flrtvc-and-important-improvements

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2025/08/01/important-numbers-11s-in-july

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2025/09/06/an-weird-hobby-of-a-ibm-power-champion

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2021/03/27/live-update-cookbook

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/06/25/aix-advanced-rbac-part1

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/08/03/aix-advanced-rbac-part-2

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/09/05/aix-advanced-rbac-part-3

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/09/27/aix-advanced-rbac-part-4

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/05/22/dsc-chsignpolicy

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/02/08/aix-and-te-sec-part1

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/02/22/aix-and-te-trusted-execution-an-underestimated-sec part2

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/03/21/aix-and-te-trusted-execution-an-underestimated-sec part3

https://community.ibm.com/community/user/power/blogs/christian-sonnemans1/2024/04/15/aix-and-te-trusted-execution-an-underestimated-sec part4

blogs to read this year 2026:

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2026/05/29/ipsec-ip-filter-issue-on-73-tl04-sp00

https://www.linkedin.com/pulse/aix-combo-patches-aka-multifixes-challenges-christian-sonnemans-3yfre

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2026/02/05/vio-fc-quirks-of-lku-and-aix-73-tl0304

https://community.ibm.com/community/user/blogs/christian-sonnemans1/2026/01/16/aix-73-tl04-upgrade-and-first-test-results-for-lku