FULL DECO PHASE I & REPURPOSE PHASE 1

Perl Automation  
Requirements/Design Document

FULL DECO PHASE I & REPURPOSE PHASE 1

Version 1.0

2016-05-09

Revision History

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Version | Effective Date (DD/MM/YYYY) | Brief Description of Change | Affected Section(s) | Prepared By | Reviewed By | KW Procedure Revision |
| 1.0 | 2016-05-09 |  |  | Pradeep kumar Ganjur Nagaraj | cashisea | R8 |
|  |  |  |  |  |  |  |

Table of Contents

1. Introduction 4

2. Out of Scope 4

3. Assumptions 4

4. Business Requirements 5

4.1 Prerequisites 5-10

4.2 Preparation 11-19

4.3 Deployment 20-23

4.4 Checkout 25

4.5 Backout………………………………………………………………………………………………..………………………………………………26

5. Use Cases 27

6. Automaton Design 27

6.1 Automaton Diagram 27-29

6.2 Automaton Steps 30

6.3 Automaton Dependencies 30

**7. Error Handling 30**

**8. Risks 30**

9. Testing Approach 30

9.1 Unit Testing 30

9.2 System Testing 30

9.3 UAT Testing 30

10. Future Development 31

# Introduction

This document describes the requirements and design of the new automaton that will be created to automate the steps performed in the Knowledge Web procedure titled FULL DECO PHASE I & REPURPOSE PHASE 1.

Automating this Knowledge Web procedure will greatly reduce the amount of effort that support personnel spend addressing issues of this type.

*Project does the phase1 decommission of the host; basically we are supposed to check the server status as “build” and proceed with decommission. If the host status is other than build we need to gather approvals and proceed with the phase1 deco.*

*Script will remove all the dependencies from the host and proceed to phase2 decommission*

*Finally we have set the host status to decommission and move to phase2 deco*

The Knowledge Web procedure can be found here: <http://wiki-na.ms.com/KnowledgeWeb2009/LinuxUnix/RepurposeServerDecoPhaseI> & <http://wiki-na.ms.com/KnowledgeWeb2009/LinuxUnix/FullDeco>

This document applies to the version of the Knowledge Web procedure modified on 28 Jan 2015.

# Out of Scope

* Any process not explicitly mentioned in this KW document.

Automatic TCM ticket creation

# Assumptions

* It is assumed that the Knowledge Web procedure will not change within the duration of this project. However, if the operational team deems that a change to the procedure is necessary, operational team will communicate that proposed change to the WPI team so that an assessment can be made on the impact to this project.
* It is assumed that a proid will be given access to run all the commands listed in the Knowledge Web procedure.

# Business Requirements

## Prerequisites

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Procedure Step** | **Sub-Step** | **Business Requirement** | **Automation Approach** |
| **1** | MIS/GAMI Ticket - All requests for REPURPOSE and DECO requests must come through MIS |  |  | This step is not going to be automated. |
|  |
| The script will check is there any information in the TAS Logger relevant to the host. If there is no information than the script will proceed with gathering the information of the host. Otherwise the script will parse the output of the logger and proceed with the appropriate phase. |
| **2** | Host Status (as per servinfo) |  | Perl | In the Perl script perform command: |
|  |  |
| Build - MCM SA's will manage the work - approval requirements and notifications are exempt | servinfo <*hostName*> |
|  |  |
| All other status - Approvals and notifications as documented in the procedure are required | From the result of the command retrieve the value of Status field. If the value is "*Build"* then proceed otherwise investigate if the host is required any notifications and approvals before proceed. If we need approvals we will ask RMC to receive it at the end of the processes of gathering host details |
| **3** | **Production DB hosts requires DBHostAuthorizations procedure** | [Procedure](http://wiki-na.ms.com/KnowledgeWeb2009/LinuxUnix/DBHostAuthorizations) |  | This step will perform if the host is a DB host. This procedure is partially automated. See below for details. |
| **3.1** |  | **Prerequisites** | Perl | The status of the host has been identified at Step 2. The region can be determined by performing following command: |
| Procedure initiated from Full Deco or RepurposeServerDecoPhaseI |  |
|  | servinfo -s <*hostname*> -g loc:region -raw |
| Hosts status is "Production" |  |
|  | For the result of the command perform search with the following values: |
| Determine region(s) affected by change(s) |  |
|  | North America |
| All other approvals necessary for the work | Asia Pacific |
|  | Europe |
| **3.2** |  | **Preparation** |  | By this step we've already known from the step 2 that: |
| If the ESP is not Production, continue with the deco process (DBA approval not required) | The work type is Decommission or Repurpose |
|  | The status of the host if Production |
| Check work types to determine if DB Authorization is required: | The request comes through MIS |
|  |  |
| Work Types that require approval | So we've everything that we need to proceed to the deployment stage |
|  |  |
| Decommission of host |  |
|  |  |
| Tasks normally covered by GAMI, but being pushed through as an emergency |  |
|  |  |
| If the Work Type is unclear, or you are not sure if DB Authorization is required, contact a Senior SA or L3 for clarification |  |
|  |  |
| If the Work Type requires approval, continue with this procedure |  |
| **3.3** |  | **Deployment** |  | From the step 3.1 I've known the region of the host. The script will be using only primary contact from the appropriate region for an escalation. An escalation to the secondary contact or the emergency contact is not going to be automated and should be performed manually if needed. |
| Determine the approver(s) for the region in which the change is to effect: |
|  |
| North America: |
| Primary: Kendra.Floyd (kendra) ED +1 212 276-0320 NY-1NYP/09 |
| Secondary: Joseph.Rooney (joro) ED +1 212 276-0246 NY-1NYP/09 |
|  |
| Asia Pacific: |
| Primary: Ming.Li (mingli) ED +44 20 7677-3614 LN-CS/06 |
| Secondary: Motohiro.Shimizu (mshimizu) VP +81 3 5424-4452 TK-EBISU/08 |
| Secondary: Catherine.Ip (cathip) VP +852 3963-2624 HK ICC/30 ISG/IT |
|  |
| Europe: |
| Primary: Ming.Li (mingli) ED +44 20 7677-3614 LN-CS/06 |
|  |
| Escalation: Steve.Russell (stevenru) MD +1 212 276-0484 NY-1NYP/09 (this is if you can not reach any of the specified points of contact for approval) |
| **3.4** |  | **Escalations** | Perl | At the end of the process of gathering host details the script will ask would you like to escalate. For the process of escalation the script will ask the user to provide the customer email as well as the ticket number. Brief description of the change will be following: |
| Reason and Contact Methods: |  |
|  | "repurpose of the host <*hostname*>" or |
| Send an email to the approver as determined above. Be sure to include the customer on this mail thread as well as DBAU. Ensure mail group <dsops> is also on the email, and subject contains the ticket number and brief description. | "decommission of the host <*hostname*>" |
|  |  |
| The mail should read something similar to the following: |  |
| Sample Email to DB approver |  |
| <Approver> |  |
| The following request <ticket number> is requesting <brief description of the change>. Prior to doing this work, please review the request and notify the RMC if we can proceed with this request. |  |
| Thank you, |  |
| <RMC Signature> |  |
|  |  |
| Once the Email approval has been given, the RMC can proceed with the customer’s request after any additional required authorizations are obtained. (Return to the correct deco procedure to complete the procedure) |  |

## Preparation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Procedure Step** | **Sub-Step** | **Business  Requirement** | **Automation Approach** |
| **1** | **Authorization and Approval** |  | Perl | Additional to the step 2 of the DBHostAuthorizations procedure that was describing earlier the script will show the following information to RMC to help to determine if the host if a production database host in case of the status of the host is not state that: |
|  |  |
| Review MIS/GAMI Ticket for Deco/Repurpose | the contact list by using: servinfo -s <*hostname*> -g con:contact\_name, contact\_role |
| If the request did not come from MIS (GAMI Ticket), instruct requestor to enter the appropriate MIS ticket at: | the description and the usage groups of the host by using: servinfo -s <*hostname*> -g sum: description, usage\_groups |
| http://MISPROD |  |
|  | If that information doesn't help to determine production database host or if RMC is unsure the script will suggest interrupting a process of decommission or repurpose of the host and reach out to the RMC's team lead or L3 to clarify that. |
| If the host is a production database host, follow the DBHostAuthorizations procedure. |  |
| Is DBAU a member of the contacts list? If Yes, Follow DBHostAuthorizations procedure |  |
| Are the description / usage / hostname indicative of a database host? If Yes, Follow DBHostAuthorizations procedure |  |
| If you are unsure, reach out to your team lead for suggestion. You may need to escalate to L3. |  |
| Once all necessary approvals have been received (MIS, DB authorization if DB host) proceed to the Phase 1 Prep Procedure. |  |
| **2** | Check the server database for critical information: servinfo hostname |  |  | The script will show the output of the command servinfo witch contains all names of the host as well. |
|  |  |
| Determine all NAMES (host names and aliases, including service groups if any) | At this step script will determine the hardware via servinfo with key "hw" |
|  | servinfo -s <hostname> -g hw |
| Determine the hardware make and model of the server via servinfo |  |
| For Sun server models — Netra T-1, E420, E4500, E6500, 280's, and higher — please send an email to ec-integration-na@ms.com or ec-integration-eu@ms.com before weekend work | At the end of the process of gathering host details in case of Sun server the script will ask would the user like to send the email to appropriate ec-integration. |
|  |  |
| If we are re-using this hardware, make a note of that in the Phase 2 Deco request. DCIM should not be instructed to remove the hardware after Phase 2 is complete. EC-integration will change the server allocation in ESP to "inventory pool." |  |
| **3** | Unless the host has status "Build" confirm the console is working for the host |  |  | In case of the status of the host is not "Build" the script will perform the command: |
|  | *bsm -a set -t passwd -n <hostname>* |
| Unless the host has status "Build" confirm the OOB password is current: bsm -a set -t passwd -n $host should return "NOTICE: OOB password successfully updated" |  |
|  | The results of the command will compare to "NOTICE: OOB passward successfully updated". If the results is not match with that than the script will exit with the appropriate message. |
| **4** | **Gathering Host Data for Weekend Work** (full procedure: http://wiki-na.ms.com/KnowledgeWeb2009/LinuxUnix/IncludeWEWGatheringHostData) |  | Perl | At this step the script will perform the command |
|  |  |
| Before you begin your work you should run decogen and specify the output dir as | */ms/dist/discovery/PROJ/decogen-harvest/prod/bin/decogen -x -d /ms/group/it/rmc/sys\_hist/<hostname>/before* |
| /ms/dist/discovery/PROJ/decogen-harvest/prod/bin/decogen -x -d /ms/group/it/rmc/sys\_hist/$hostname/before |  |
|  | After the process of decommission or repurpose of the host the second command |
| After you finish your work you should run decogen and specify the output dir as |  |
| /ms/dist/discovery/PROJ/decogen-harvest/prod/bin/decogen -x -d /ms/group/it/rmc/sys\_hist/$hostname/after | */ms/dist/discovery/PROJ/decogen-harvest/prod/bin/decogen -x -d /ms/group/it/rmc/sys\_hist/$hostname/after* |
|  |  |
| The script must be run as yourself on the host you will be working on. | have to run manually as well as to compare the result before and after the work |
|  |  |
| Data comparison |  |
|  |  |
| diff -r /ms/group/it/rmc/sys\_hist/$hostname/before /ms/group/it/rmc/sys\_hist/$hostname/after |  |
| **5** | Run checkbootnet on the server as yourself: checkbootnet hostname |  | Perl | This step will connect to the host via ssh and perform the command checkbootnet: |
|  |
| /ms/dist/aurora/bin/ssh -o StrictHostKeyChecking=No -C <hostname> /ms/dist/aurora/bin/checkbootnet <hostname> |
|  |
| The output of the command will include to the log file witch will be sent to performer of the script via email and showed in the console. |
| 6 | Determine Dependencies |  |  | **Per Cory Wright email this step does not require a new ticket to be created for escalating it to L3** |
| Determine configured dependencies for each name: Checklist for the identification of dependencies If any dependencies are found sends a task ticket to L3 to identify and remove any core infrastructure dependencies. Once this is complete, L3 will return the ticket to L2 and the RMC will contact the host owners and/or requester to obtain approval to deco the host. |  |
|  |  |
| **7** |  | **Appendix: Checklist for the Identification of Dependencies on NAME** | Perl | the script is going to build the array out of the hostname, the aliases of the host, the service groups by performing command: |
| Describes the common tasks required for deletion of a name (hostname, alias, service group) |  |
| The following steps are performed for each name being decommissioned | *servinfo -s <hostname> -g alias* |
| Find and document all dependencies on NAME in DSDB, OSCONF, and so on |  |
|  | The result of the command will be parsed in order to create the array of dependencies |
| **8** |  | dsdb show\_host -host NAME should not show any host aliases | Perl | For the each item in the array from the previous step perform |
| dsdb show\_host\_dependencies -host\_name NAME |  |
|  | *dsdb show\_host -host <itemName>* |
|  | *dsdb show\_host\_dependencies -host\_name <itemName>* |
|  |  |
|  | the result will be sent to performer of the script and showed in the console |
| **9** |  | On the host run these commands: | Perl | The script will connect to the host via SSH and will get the content of the files in order to search the name of the host: |
| grep NAME |  |
| /etc/bldg/conf/podconfig/\* /etc/campus/conf/podconfig/\* /etc/pod/root/etc/\*.conf /etc/pod/root/etc/\*.cf /etc/campus/etc/\*.cf /etc/region/root/etc/\*.cf /etc/bldg/etc/noreboot /etc/bldg/etc/mslinks/devaccess /etc/bldg/etc/mslinks/rwaccess /etc/pod/root/etc/mslinks /etc/pod/root/etc/devaccess /etc/pod/root/etc/printcap /ms/dist/kerberos/PROJ/keytabmgr/incr/common/etc/new\_dst.cf /ms/dist/aurora/etc/flexlm.conf /ms/dist/aurora/PROJ/dns/incr/etc/secondaries.pl | *$SSH = '/ms/dist/aurora/bin/ssh -o StrictHostKeyChecking=No -C';* |
|  | *$res = `$SSH $host "cat <eachListedFile>" 2>/dev/null`;* |
|  | *grep /\b<hostname>\b/, $res;* |
|  |  |
|  | The results of the search will be sent out to the performer of the script via email. |
| **10** |  | krb5\_prestash query host NAME |  | The script will perform the command: |
| If this command produces a list of proids, you need to contact the owners one-by-one and ask them to delete their proid from the host you`ve just decommissioned. You can find out the owner of the proid, if you run the following command: | *krb5\_prestash query host <hostname>* |
| proid query $proid|grep ^owner |  |
|  | At the end of the process determine dependencies if any proids will come up than the script will ask would the user like to send an email to the owners of the proids with a request to delete the proids. |
| If they do not know the way to delete the proid, provide the following command to them: |  |
| krb5\_prestash remove $proid $hostname.ms.com (example: krb5\_prestash remove r1user oz36c2n14.ms.com) | As well in case of repurpose the script will skip \_autach\_ proid. |
|  |  |
| Also provide them the following link for further details on prestash tickets. This will send the users to the right direction in case of any issues removing their proids from the host: http://secwebapp.ms.com/kerberos/docs/products.krb5\_prestash.html |  |
| Note In case of repurpose the only proid you must not delete is the: \_aubatch\_ |  |
| **11** |  | Dependencies on Autosys (must be run from an NA box): /ms/dist/unixops/PROJ/rmc/incr/bin/alias.pl \_NAME\_ (only needs to be run using the primary host name) | Perl | If the script has been started from non-NA environment then the script will notify the user and skip this step. In case of NA environment the script will perform the command: |
|  |
| /ms/dist/unixops/PROJ/rmc/incr/bin/alias.pl <hostname> |
|  |
| The result of the command will be sent out to the performer of the script via email |
| **12** |  | DB server dependencies: | Perl | For the each entry in the output of the command: |
|  | srvloc -host <hostname>services |
| for i in NAME `srvloc -host NAME services`; |  |
| do /ms/dist/syb/dba/scripts/ssl -aw $i; | perform the command: |
| done | grep -v OP\_ | /ms/dist/syb/dba/scripts/ssl -aw <entry> |
|  |  |
| /ms/dist/syb/dba/scripts/asl -H \_NAME\_ | for the result perform: |
|  | grep -v OP\_ |
|  |  |
|  | The output of the command will write to the file and sent to the performer of the script via email, as well as the output of the command: |
|  | /ms/dist/syb/dba/scripts/asl -H <hostname> |
| **13** |  | Shared SAN dependencies: run /ms/dist/unixops/PROJ/storagetools/2.0/bin/sandetach $host... > /dev/null and handle any errors as follows | Perl | The script will perform the command: |
| "host $host was not found in database" - ignore - host is not SAN-attached | */ms/dist/unixops/PROJ/storagetools/2.0/bin/sandetach <hostname> > /dev/null* |
| "more hosts in zone than specified" - other hosts share storage with this host - confirm with the requester if no more hosts have been specified or specify them on the command line |  |
| "too many hosts specified" - you have specified unrelated hosts on the command line | the result will be sent to performer of the script and showed in the console |
| **14** |  | **HLM Dependencies** | Perl | The script will perform: |
| Please remove hlm dependencies (if any) from the host | *module load laf/hlm/prod* |
| module load laf/hlm/prod | *hlm get <hostname>* |
| Get the hlm data associated with the host to be decommissioned hlm get $host |  |
| Warning: Never ask user or delete HLM Login Privileges found during prep. | For the output of the command |
| Delete the netgroups if any. hlm remove $host –netgroups $netgroup | *hlm remove <hostname> –netgroups <netgroup>* |
| Delete the host from hlm. hlm delete $host |  |
|  | Perform the command: |
|  | *hlm delete <hostname>* |
| 15 | Notice of Decommission (for Repurpose) |  | Perl | The script will connect to the host via SSH and get all of users who logged in over the last 30 days via last command. As well the script will get host contacts via servinfo command. To all of these contacts will be send the notice of deco. |
| E-mail all ESP contacts as well as all users who logged in over the last 30 days that "As per Falcon Ticket... (Falcon ticket number) this server will be Decommissioned (for repurpose) on ... (date)". You can use the script /ms/dist/unixops/PROJ/rmc/incr/bin/phase1deco -startprepwork -host hostname script for this. |
|  |
| Update motd: Add the line "this server will be Decommissioned (for repurpose) on ... 'date'") to /etc/motd.local and run /etc/init.d/rc.motd. You can use the script /ms/dist/unixops/PROJ/rmc/incr/bin/phase1deco -startprepwork -host hostname script for this. |
| **16** | **Raise TCM for execution of Repurpose/Deco Phase I** |  |  | This step is not automatable. |
| Once Owner approval is received (for app dependency removal), the RMC SA will raise a TCM for Repurpose/Deco Phase I for Friday. |  |
| Note: Dependencies will be removed during Repurpose Phase II. | By this step script has collected all of the necessary information about the script and all of the approvals have received. Before proceeding with the deployment phase the script will write the following information using TAS Logger and exit: |
|  |  |
|  | The primary name of the host |
|  | Decommission or Repurpose of the host is performing |
|  | The steps of the gathering host information phase and the getting approvals phase have done |
|  | The date after we are ready to proceed with deployment phase (Friday) |

## Deployment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Procedure Step** | **Sub-Step** | **Business Requirement** | **Automation Approach** |
| **1** | Confirm the console is working for the host. Unless the host has status "Build" do not proceed with the deco unless the console is working.  Confirm the OOB password is current: bsm -a set -t passwd -n $host should return "NOTICE: OOB password successfully updated". Unless the host has status "Build" do not proceed with the deco unless the OBB password has been updated. |  |  | The step to automate this.  The script will check the information from the TAS Logger in order to ensure that we are ready to deployment:  The current date is further that is in the logger Gathering the information of the host and getting approvals are done The record in the logger is match with what the user intend to do (deco or repurpose)  In case of any mismatch the sript will notify the user and exit |
| **2** | Blackout machine for 48 hours: blackout hostname 48:00 "deco - TCM 6XXX"  Run the command: /ms/dist/unixops/bin/hoststatus -host $hostname -status Decommissioned This will update the DSDB and ESP with the Decommissioned status. |  | Perl | Perform the command in the script: `/ms/dist/aurora/bin/blackout <hostname> 48:00 \"deco TCM 6XXX\"` where 6XXX is ticket's number that has been gotten earlier  Perform the command in the script: `/ms/dist/unixops/bin/hoststatus -host <hostname> -status \"Decommissioned\"` |
| **3** | Record the Falcon request ticket, console information, IP address, MAC address, and SAN ports in ESP under the "Key Attributes" section in the "Notes" field |  | Perl | This step has been already automated by the script /ms/dist/unixops/PROJ/rmc/incr/bin/phase1deco. The function update\_fields collect all of necessary information and update ESP using /ms/dist/aurora/bin/servupdate command. We can just move this function to our script in order to do not perform phase starttcmwork that probably do a slightly different job. |
| **4** | Record from ESP the following additional information:  Node Name Switch port information Console port information |  | Perl | To get a host's console entry  dsdb show\_remprog\_member -member\_name [hostname]-cons  The output will be parset in order to get Node name and Console port.  NOTE: The only info that is not gathered is the switch port info, the script skips that info and we does not record it (according to Steve Allen). |
| **5** | **Halt the machine and disable booting:**  For Solaris: Halt the host and reset the boot device with deco message, boot-device, date, & SA name Use printenv boot-device at the ok prompt or eeprom boot-device at the OS level to see the current device. The value will be used to substitute the previous\_boot\_device\_info below. Change it to the same thing with a prefix: setenv boot-device deco\_<mm-dd-yyyy>\_<yourid>\_<previous\_boot\_device\_info>  For Linux: Run /etc/init.d/rc.update -force -noreboot Update /boot/grub/grub.conf so that deco\_<mm-dd-yyyy>\_<yourid> appears after the word kernel on line that define the kernel to boot with (For Example: kernel deco\_06-10-2005\_gopipra /boot/vmlinuz-2.4.9-e.3 ro root=/dev/sda1). Halt the host |  | Perl | For Solaris: The script will build the deco message in the following format: $new\_boot = "deco\_".$date."\_".$user ID."\_".$current\_boot\_device   and offer to the user perform the command mually: "Execute \"eeprom boot-device=$new\_boot\" \n to change the boot device of the host $host and halt the machine."  For Linux: The script will build the deco message in the following format: $new\_boot = "deco\_".$date."\_".$user ID and offer to the user perform the command mually:  "Execute \"/etc/init.d/rc.update -force -noreboot\" and update /boot/grub/grub.conf so that $new\_boot appears after the word kernel on line that define the kernel to boot and then halt the machine." |
| **6** | If VCS cluster, set configs to decommissioned:  module load vcs msvcs status -update decommission -cluster <clusterID> gmake <clusterID> dist |  |  | The script will determine if the host is a cluster and offer to the user perform the commands manually in case of cluster |
| **7** | Update Falcon ticket (if any) to include notes that the Repurpose Phase I was completed and update the brief description to say "Repurpose Phase II ."  Raise a TCM for the Repurpose Phase II for the following weekend. You may need to add it: The TCM flag Post-Event Entry should always be set to NO Make a notation in the TCM description: Info Note: This is a Repurpose Phase II : Do NOT remove DRAC, console info, box DSDB entry. Include the Repurpose Phase I TCM number in the Repurpose Phase II TCM request's "Linked TCM Requests" field.  For all Repurpose Phase II scheduling — If the Repurpose Phase I ticket specifies a different date (For Example, two weeks or a month out), schedule it for the user-requested date. Update Falcon ticket (if any) to include notes that the Phase I Deco was completed and update the brief description to say "Phase 2 Deco."  Raise a TCM for the Phase II Deco for the following week. You may need to add it: The TCM flag Post-Event Entry should always be set to NO Include the Phase I Deco TCM number in the Phase II Deco TCM request's "Linked TCM Requests" field For all Phase II Deco scheduling — If the Phase I Deco ticket specifies a different date (For Example: Two weeks or a month out), schedule it for the user-requested date. |  |  | This step is not automatable. |

## Checkout

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Procedure Step** | **Sub-Step** | **Business Requirement** | **Automation Approach** |
| **1** | Host is down |  |  | This step is not automatable |
| **2** | Check that host is blacked out |  | Perl | To check that the host is blacked out the script perform the command: *`/ms/dist/aurora/bin/blackout -c $host`*  We will parse the output of the command in order to check that the host is blacked out |
| **3** | Host status has been set to "decommission" |  | Perl | In the Perl script performs command:  servinfo <*hostname*>  From the result of the command retrieve the value of Status field and compare the value to "decommission". |
| **4** | Phase II TCM has been raised |  |  | This step is not automatable  By this step script has deployed decommission or repurpose of the host. The script will write the following information using TAS Logger and exit:  The primary name of the host Decommission or Repurpose of the host has been performed |

## Blackout

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Procedure Step** | **Sub-Step** | **Business Requirement** | **Automation Approach** |
| **1** | If a customer asks to have a server undecoed you need to get approval from regional management. The request should be coming from the box owner. |  |  | This step is not automatable |
| **2** | If you cannot reach regional management within 30 minutes of the request to undeco and proceed with undecoing the host and send email notification to regional management, Cc the weekend notification contacts (usually UnixOps for the region, keane-rmc-runbook@ms.com, ec-integration@ms.com). To-Undeco, undo the changes performed. |  |  | This step is not going to be automated due to each situation is very specific. |
| **4** | **Blackout Steps Method-1**  Confirm mconsole is working. Powercycle the host. In the Grub, boot the host from the second kernel, as the default kernel will be updated with "deco\_<mm-dd-yyyy>\_<yourid>" Login to the host (once it has booted) & edit the file "/boot/grub/grub.conf" and remove the entry "deco\_<mm-dd-yyyy>\_<yourid>" from the first kernel. Powercycle the host to confirm it boots properly. |  |  | This step is not automatable |
| **5** | Signoff the host as per customer's request using ServerSignoff procedure. |  |  | This step is poor automatable. So there is no reason to automate this |
| **6** | Unblackout the host if it is still in blackout using the command /ms/dist/entmgt/bin/unblackout -host $host |  |  | This is automatable but the rest of the steps aren’t. So there is no reason to automate this. |
| **7** | **Method-2 (if console is not available)** Run spoti install\_os -host $host Signoff the host as per customer's request using ServerSignoff procedure. Unblackout the host if it is still in blackout using the command /ms/dist/entmgt/bin/unblackout -host $host If it is a cluster, signoff the VCS using the VcsSignoff procedure. |  |  | This step is poor automatable because of Signoff procedure. So there is no reason to automate this |

# Use cases

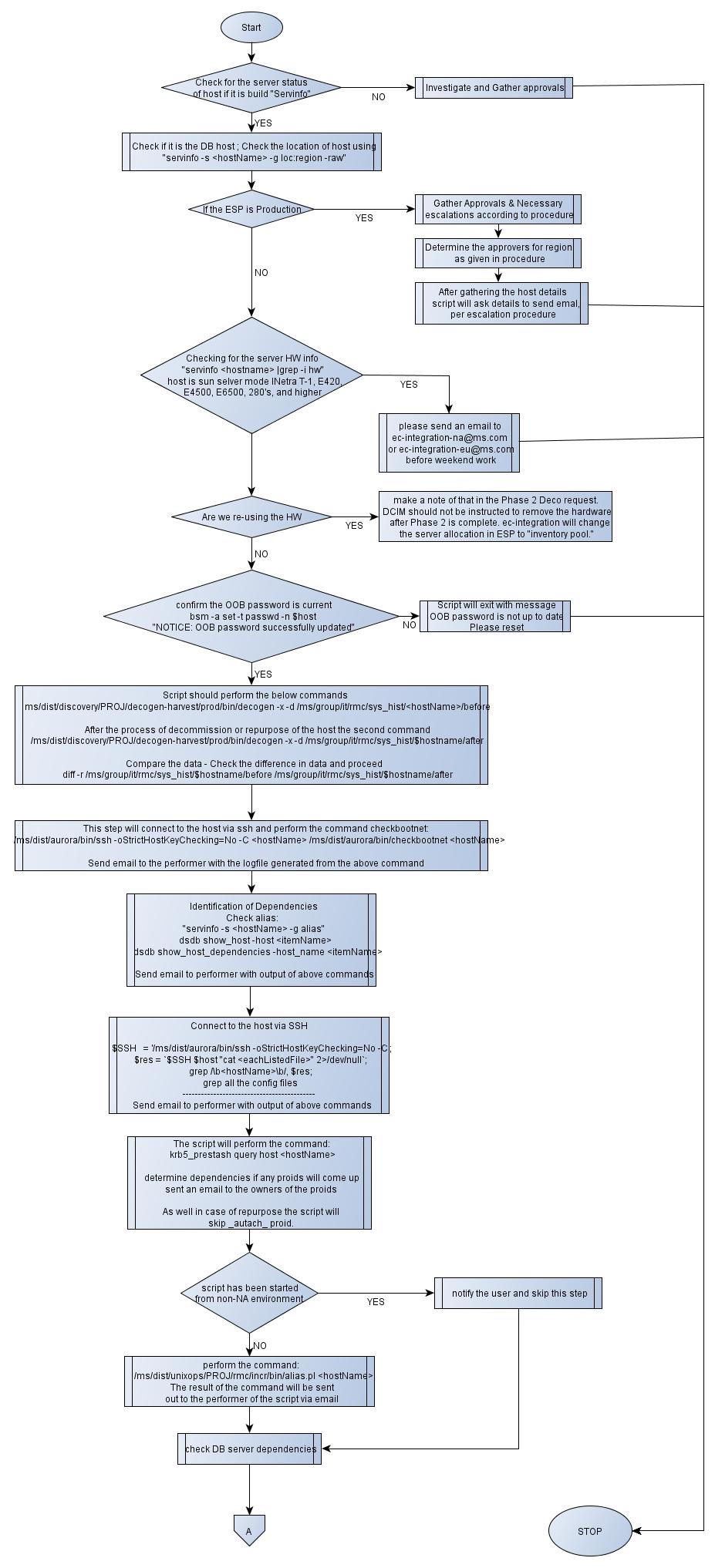
A use case is to identify, a task or goal that users will be able to accomplish using a program. It should be written in terms that users can understand. Provide all of the script available functionalities.

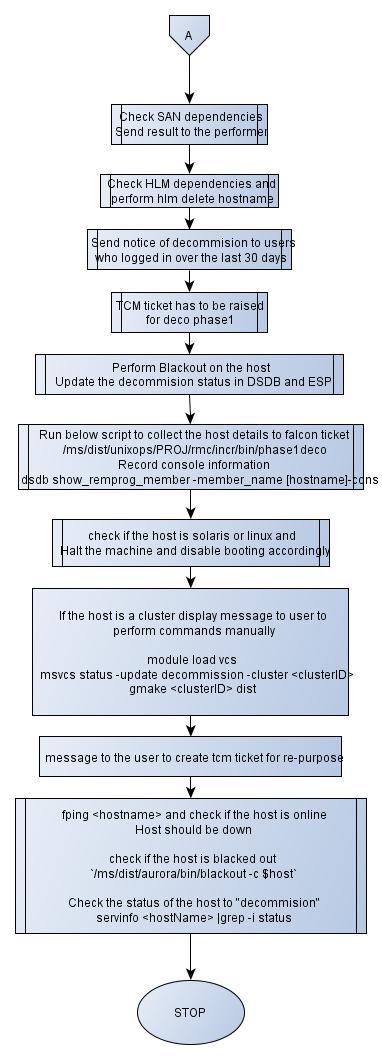
# Automaton Design

## Automaton Diagram

The following is a diagram of the proposed flow of the new automaton. A detailed description of each state and transition will be described in the next section.

Please find the flowchart in the next page





## Automaton Steps

*Project does the phase1 decommission of the host; basically we are supposed to check the server status as build and proceed with decommission.*

*Script will remove all the dependencies from the host and proceed to phase2 decommission*

*Finally we have set the host status to decommission and move to phase2 deco*

## Automaton Dependencies

*Automation is dependent on the DSDB and ESP information of the host and the service now TCM ticket creation*

# Error Handling

If any error occurs during execution of the automation, the appropriate error message will be stored in the log file and print on the screen.

*Message is displayed to the user at many instances to gather approvals*

# Risks

* If proids are not granted access to required commands or test environment in a timely manner, there is a risk that the project will be delayed.

# Testing Approach

## Unit Testing

The person conducting the development of these automata will also conduct Unit Testing. A formal Unit Test plan will not be used.

## System Testing

A formal System Test plan will be created and executed by a developer on the WPI team. All logical paths, described above, will be thoroughly tested. Each test case will examine all the flows mentioned above. All error conditions will be tested.

## UAT Testing

After the unit and system testing is completed, the developer would send a test plan to the stakeholder’s to test the application. If the testing satisfies the stakeholder’s expectations of the automation, they will provide their UAT sign-off via a follow-up email.

# Future Development

*Phase 2 decommission would be developed to complete the full decommission of the host*