Control Duplicate Host Creation (ADDM)

Idea
- ADDM should NOT create duplicate hosts for Servers and Workstations that have been upgraded. See Implementation details below.

Background
- Many Companies are implementing a Private Cloud environment where VMare based Hosts and Solaris Zones are continually being cloned and replaced for maintenance purposes. Also, many companies periodically upgrade their Workstation fleet from one version of Windows to another.

As a result of both these activities, ADDM's inventory data contains duplicate hosts. It can take 60 hours to run a full scan so scans may only be scheduled once a week and Nodes may only age out after a month or more. This means that sometimes ADDM may only be 90% accurate - even when model maintenance parameters are optimized. This affects downstream processes such as: (a) Automatic Change Conflict Detection; and (b) Software License Management. Try telling the Change team at a Bank that: "ADDM is only 90% accurate and you should expect 2 undetected change conflicts per year and the risk of one Sev 1 outage". What reaction will you get?

Current ADDM Solution

The current ADDM weighting table is no longer a good mechanism to control the host creation process. In a Virtual world, many of the Attributes used in the weighting table such as Serial Numbers and Zone_UUID and UID have little meaning or change completely when a host is cloned.

Solution

If it is possible to detect that a host is the exactly the same or is a direct replacement, then ADDM should make every attempt to ensure that a duplicate host is **not** created. User's should be given control of the weighting table, and new attributes should be included (see below).

The weighting table works by calculating a score when deciding if to create a new Host Node. If the score is positive, a new Node is created. If the score is zero or negative, the existing Host Node is updated.

Possible Implementation

I would make the following changes:
- Make the ADDM Host Weighting Table visible and configurable by the user.
- Add IP-Addresses and Filesystems to the list.
- Differentiate between Physical and Virtual MAC Addresses.

- Set the following default values:

#	attribute	Identical	Some different items (lists only)	Completely Different	New Value is Null
1	hostname	5	N/A	-20	0
2	Physical MAC Addresses	5	-5	-20	0
3	Virtual MAC Addresses	1	-1	-1	0
4	ram	1	N/A	-1	0
5	os	1	N/A	-1	0
6	num_logical_processors	1	N/A	-1	0
7	Filesystems	1	-2	-20	0
8	dns_domain	1	N/A	-1	0
9	IP-Addresses	1	-2	-20	0
10	Processor Type	1	N/A	-5	0

Use Cases

#	Scenario	Attribute (above) that change	Total Score	New Node Created?
1	A Physical Windows Desktop is updated from XP to Windows 7. (serial # and uuid would change)	os	+11-1 = 10	No
2	A Solaris Zone is cloned and moved from the PROD environment to the TEST environment to conduct some testing. During the transfer, the	Virtual MAC Addresses, IP-Addresses	+11-21 =-10	Yes

	IP-Addresses are changed.			
3	A production zone is cloned, temporarily moved, upgraded and then moved back into Production replacing the old zone.	Virtual MAC Addresses	+12-1 = 11	No
4	A User is given a new Lenovo laptop to replace his old Dell. The new machine has the same hostname.	Physical MAC Address, processor type.	+6-25 = -19	Yes
5	The UNIX team deploy two identical virtual Linux Hosts on VMware as a pair of HA Web Servers.	hostname, Virtual MAC Addresses and IP Addresses.	+6 -41 = -35	Yes
6	Some DR Examples - both cold and hot DR.	??	??	Not Sure
7	Back-Out, A admin backs out a change by starting the cloned copy created just before the change	Virtual MAC Addresses,	+12-1=11	No
8	A Physical Solaris server is upgraded. all the Zones are migreated to hte new box	I think nothing changes	+13	No