

Requirements Specification

Colorado

*Andy Hisgen
Thorsten Früauf
Nick Solter*

June 3, 2008

FBCs 2008/1717 and 2008/1718

Revision History

<i>Version</i>	<i>Comments</i>	<i>Date</i>	<i>Author</i>
1.0	Initial draft	06/03/2008	Andy Hisgen
2.0	Added Colorado requirements	6/20/08	Nick Solter
2.1	Updated after discussion with Andy and Thorsten	6/25/08	Nick Solter
2.2	Updates after colorado dev meeting	7/2/08	Nick Solter
2.3	Comments from Ed, Andy, and others	8/12/08	Nick Solter
2.4	Another revision	8/21/08	Nick Solter
2.5	Added Samba to list of agents for phase 1	9/15/08	Nick Solter
2.5	Revisions per first day/written comments of CLARC inception	10/1/08	Nick Solter

Always update the revision history after making changes to this document.

1. Project Description

1.1 What it is

Project Colorado is a binary distribution of Sun Cluster that runs on the OpenSolaris binary distribution[1]. There are two main thrusts to this project:

1. Porting Sun Cluster to the OpenSolaris distribution, including the new Image Packaging System[2].
2. A minimal and extensible cluster, which provides basic cluster functionality with a low barrier to entry and easy deployment for OpenSolaris. The intended audience is system administrators needing [simpleeasily configured](#) HA and developers needing an HA framework for their application, both inside and outside of Sun.

1.2 *Why Colorado?*

As described in the previous section, the Colorado project can be thought of as two projects in one. The first thrust of the project is to run Sun Cluster on OpenSolaris. As described in section 1.3 below, the OpenSolaris distribution is significantly different from Solaris Express, Solaris 10, and any other previous Solaris release. Thus, it's not just a "compile and run" exercise to enable Sun Cluster on OpenSolaris; it's really a project amount of work.

The goal of the second thrust of the project, to provide a minimal and extensible cluster, is to counter the perception, and to some extent reality, that Sun Cluster is an unwieldy, monolithic, and difficult to setup, configure, and administer software system. By reducing the barrier to entry in both the hardware and software areas, this project will enable Sun Cluster to be used in areas in which it was not previously considered.

1.3 *Background on OpenSolaris and IPS*

OpenSolaris is the binary distribution of the OpenSolaris source code base. Starting with OpenSolaris 2008.05, OpenSolaris will be released on a six-month cycle, with an 18-month support cycle.

OpenSolaris is distributed as a LiveCD, from which the user can install a minimal environment, including the GNOME desktop and other desktop tools (Firefox, etc). Additional packages can be obtained from network package repositories. The main repository currently is pkg.opensolaris.org. Others include pkg.sunfreeware.com and blastwave.network.com. The pkg.opensolaris.org repository can contain only freely redistributable binaries. A proposed repository, pkg.sun.com, will be able to contain encumbered and non fully redistributable binaries.

OpenSolaris has replaced SVR4 packages with the new Image Packaging System (IPS). IPS packages do not allow scripting in the form of preinstall, postinstall, etc. scripts.

Upgrades in OpenSolaris use the concept of "boot environments," which depend on ZFS snapshotting and are managed with a new beadm command. A full upgrade automatically creates a new boot environment. The [useradministrator](#) can rollback to the old BE at any time, and can create new BEs at any time.

Other changes in OpenSolaris include ZFS as the root file system, ksh93 instead of ksh88, bash as the default user shell, and much new functionality (such as NWAM) that is not present in Solaris 10.

1.4 *Scope*

1.4.1 *Smallness defined*

Colorado is treating smallness as consisting of two attributes:

1. Hardware Minimization (e.g. the ability to run without special hardware fewer supplementary hardware requirements)

Colorado runscan run on stock off-the-shelf hardware. The ability to run on stock hardware will make Colorado more attractive to system administrators and developers and also suitable to more

uses, e.g., software appliances. Colorado removes the historical Sun Cluster requirements for ~~special~~supplementary hardware; in particular, private interconnects and shared disk storage become optional components. When shared disks are present, Colorado will leverage the optional fencing project which is part of SC3.2u2.

2. Software Minimization

Colorado provides a subset of the complete Sun Cluster functionality. ~~It leverages the IPS packaging system to~~ by ~~makeing~~ many components optional. Colorado will thus be delivered as a set of core versus optional IPS packages, thus reducing both the static and dynamic footprint, including:

- Size of the core packages on distribution media or as download.
- Size on disk.
- Runtime size in memory and in swap, both kernel and user-space, and including user-space demon processes.
- CPU consumption at runtime.

1.4.2 Phased Approach and Phases

The overall Colorado project will consist of three phases. This section is easier to understand after reading the requirements below. Each requirement below lists the Phase in which it will be implemented.

Note: Only the requirements for Phase 1 are “finalized” at this time. Requirements for phases 2 and 3 will be reassessed after Phase 1 is complete. Specifically, the alternatives for hardware minimization may or may not be included in phases 2 and 3.

Phase 1

Executive Summary:

Colorado Phase I will provide a fully functional release of Sun Cluster on OpenSolaris 2009.04 on x86 that can run on stock hardware without private interconnects, shared storage, or quorum device.

Details:

Phase 1 will align with OpenSolaris 2009.04. It will include:

- Porting Sun Cluster to OpenSolaris, including IPS. However, phase I will not include binaries built from encumbered code or SPARC support. It will also not include geo edition.
- Agent support will focus on open source applications that run on OpenSolaris.
- Hardware Minimization
 1. The private network(s) requirement will be eliminated (~~note: dependent on Crossbow project~~).
 2. iSCSI support for local storage.
 3. Removal of quorum device requirement for two-node cluster.

- Points 2 and 3 together allow a two-node cluster to be built with any two PCs, workstations, laptops, or whatever's available, without requiring a third multi-homed storage device or quorum server.
- Zones as virtual nodes will not be supported.
 - Building with SunStudioExpress will not be supported.

Note that porting Sun Cluster to OpenSolaris and IPS is the only “must-have” requirement to ship the project. The hardware minimization aspects could be dropped if they are unable to meet the schedule.

Phase 2

Colorado Phase 2 will provide the functionality, such as zones support, missing from Phase I. Phase 2 will run on OpenSolaris 2009.10. Phase 2 will also include AVS-SNDR for using local storage instead of shared disks.

Phase 3

Colorado Phase 3 will complete the project by focusing on Software Minimization to make many more cluster components optional, to further reduce the runtime footprint, to include SPARC support, and to fully support embedding in software appliances. Phase 3 will run on the Long Term Support release of OpenSolaris, possibly OpenSolaris 2010.04.

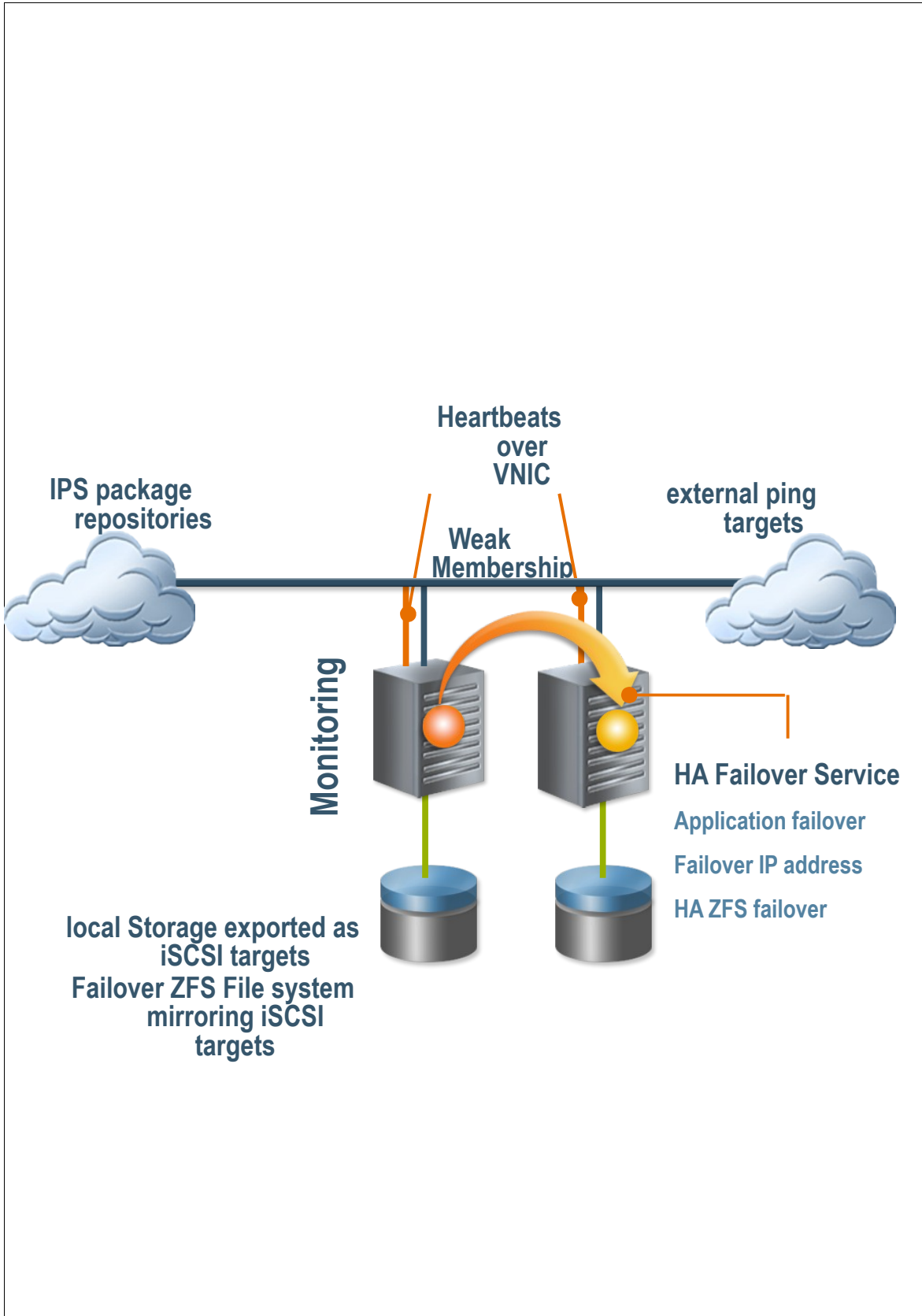
1.5 Not in Scope

The following are explicitly not goals or requirements of project Colorado:

1. Administrative smallness; reducing the amount of administration required for the cluster.
2. Installing the cluster or adding optional functionality without requiring reboots.

2. Functional Requirements

2.1 *Component Interaction Diagram*



2.2 Normal Case Behavior

2.2.1 Platform and Dependencies

- R1: [Phase 1] Colorado must run on the OpenSolaris binary distribution. Specifically, it must run on an OpenSolaris image consisting of the initial Live CD installation plus packages available from the package repositories pkg.opensolaris.org and pkg.sun.com¹.
- R2: [Phase 3] Sun Cluster Geo Edition must run on OpenSolaris.
- R3: [Phase 1] A ~~user~~ administrator must be able to install a functional cluster on an OpenSolaris image built with packages only from pkg.opensolaris.org.
- R4: [Phase 1] ~~Any~~ Cluster components that ~~is~~are dependent on Solaris/OpenSolaris components or layered software that is not available in pkg.opensolaris.org will ~~be optional~~be omitted. ~~Known components that fall into this category include, including:~~ adminconsole, SCM, ~~the GUI-based dsconfig wizards,~~ and the SunMC cluster module / SNMP module.
- R5: ~~[Phase 3] Essential functionality dependent on functionality not available in pkg.opensolaris.org will be replaced with similar components without those dependencies.~~
- R6: [Phase 3] Optional cluster functionality may have dependencies on packages from pkg.sun.com.
- R7: [Phase 1] Colorado will run on ~~32 and 64 bit~~ x86/AMD64 platforms
- R8: [Phase 3] Colorado will run on SPARC platforms.
- R9: [Phase 1] Colorado will run on one to sixteen nodes.

2.2.2 Cluster Package Distribution

- R10: [Phase 1] Colorado will be delivered as a set of IPS packages in the repositories pkg.opensolaris.org and pkg.sun.com. There will be no SVr4 packages.
- R11: [Phase 1] A ~~user~~ administrator must be able to install a functional cluster with cluster packages only from pkg.opensolaris.org. All binaries built from open source cluster code will be distributed in pkg.opensolaris.org.
- R12: [Phase 3] Binaries built from encumbered code (specifically, HA-Oracle, Oracle RAC, HA-Sybase, VxFS support, VxVM support) will be distributed in pkg.sun.com
- R13: [Phase 3] Other value-add packages and bug fixes TBD might be distributed in pkg.sun.com.

¹ pkg.sun.com does not yet exist. Requirements in this area may change as the OpenSolaris plans and our understanding of them evolve. Obviously, we can only deliver packages through pkg.sun.com after it is available.

2.2.3 Installation and Configuration

R14: [Phase 1] Colorado will define several group packages, including:

R14.1: ha-cluster-small with dependencies on all required cluster packages.

R14.2: ha-cluster-full with dependencies on all cluster packages

R14.3: ha-cluster (alias for ha-cluster-small)

R15: [Phase 1] UsersAdministrators will install the minimized cluster on each machine by running:

```
# pkg install ha-cluster
```

R16: [Phase 1] There is no need for the JES installer or the package installation functionality of scinstall. The JES installer will not be included, and the package installation of scinstall will be disabled.

R17: [Phase 1] Administrators will configure the cluster with scinstall. The configuration will include a reboot of the nodes.

Because IPS packages don't support preinstall/postinstall/preremove/postremove scripts, the current package preinstall/postinstall/etc. logic must move elsewhere. Currently recognized options include new IPS actions, such as an SMF action, or execution by scinstall as part of the cluster configuration. The requirement here is simply:

R18: [Phase 1] The lack of scripting support in IPS packages shall be transparent to usersadministrators. That is, usersadmins shall not need to run install/remove logic by hand. The cluster installation or configuration will take care of it.

R19: [Phase 1] Delivered binaries, such as SMF services and kernel modules, must be non-functional until the useradministrator configures the cluster with scinstall.

R20: [Phase 1] Complete Cluster Uninstallation: UsersAdministrators must run scinstall to un-configure the cluster or cluster node before removing the packages from the cluster or cluster nodes. Note that optional functionality, described in section 2.2.4, can be removed without unconfiguring the cluster or cluster nodes.

R21: [Phase 3] Upgrades: Colorado will support both Rolling Upgrade and Quantum Leap for upgrading a cluster to subsequent releases.

R21.1: As with the current product, cluster nodes must be taken out of cluster modes before newer versions of currently installed packages are installed.

R21.2: Rolling upgrade and quantum leap shall not be supported between phases 1, 2, and 3.

R22: [Phase 3] There will be a documented upgrade path from SC 3.2 to Colorado.

2.2.4 Optional Packages

R23: [Phases 1-3] Colorado will consist of a set of core/required packages and a set of optional packages.

R24: [Phase 1] Functionality that is currently optional in Sun Cluster must continue to be optional.

R25: [Phases 1-3] Optional packages will be dependent on one or more required packages and zero or more optional packages. The packages will be structured via dependencies as a directed, possibly cyclic, graph.

There are two aspects of adding functionality: installation of the bits and enabling of the functionality. Similarly, there are two aspects to removing functionality: disabling the functionality and uninstalling the bits. This two-step process is necessary to verify that the optional packages are correctly installed on all nodes, to enable the functionality simultaneously on all nodes, and to reboot the cluster nodes, if necessary.

R26: [Phase 3] Colorado will support the addition of optional functionality

R26.1:Optional packages can be installed at any time from the package repository

R26.2: Optional functionality can be installed without requiring a complete reinstallation or reconfiguration.

R26.3: Enabling optional functionality may require a node or cluster reboot.

R27: [Phase 3] Colorado will support the removal of optional functionality

R27.1:Optional functionality can be removed without requiring a complete reinstallation or reconfiguration.

R27.2: Disabling optional functionality may require a node or cluster rebootThere are two aspects of adding functionality: installation of the bits and enabling of the functionality. Similarly, there are two aspects to removing functionality: disabling the functionality and uninstalling the bits.

R27.3:[Phase 3] Optional packages can be installed at any time from the package repository

~~R28: [Phase 3] Optional packages can be removed only if the provided functionality provided is disabled, as per R27.~~

R29:

R30: [Phase 3] Colorado will provide a framework/mechanism for enabling/disabling optional cluster functionality.

R30.1: The cluster system management tools allow the user administrator to enable/disable the functionality.

R30.2: The choice of enabled/disabled functionality will be recorded persistently

R30.3: The cluster code must check whether optional functionality is enabled or disabled before attempting to use it.

R30.4: Attempting to use currently disabled functionality must cause the cluster to issue a polite error message and not to crash or misbehave in any way.

R30.5: Optional functionality cannot be enabled/disable on a per-node basis. It is enabled/disabled for the whole cluster.

2.2.5 Agents

R31: [Phase 1] LogicalHostname, SharedAddress, and HASP will not be optional.

R32: [Phase 1] Colorado will focus on developers by supporting the following agents, all of which will be optional: Apache Webserver, Apache Tomcat, MySQL, PostgreSQL, DNS, NFS, DHCP, Kerberos, ~~and Samba~~, and Glassfish.

R33: [Phase ~~2~~1] Colorado will support ipkg zones with the Solaris Containers agent.

R34: [Phase 3] Colorado will support the following agents, all of which are optional: N1 Grid Engine, JES Web Server, ~~JES app server~~, JES MQ, HADB, JES directory server

R35: [Phase 3+] Colorado will support the remaining agents after the underlying applications are available on OpenSolaris.

2.2.6 Functionality

R36: [Phase 1] In order to avoid code forks between Colorado and SC on S9/S10, all ksh scripts delivered as part of Colorado will work on both OpenSolaris and S9/S10.

R37: [Phases 2-3] Colorado will support whatever zones are present on our target OpenSolaris release. These may be ipkg branded zones or a new form of native zones, depending on what the solaris zones team does. Colorado will support all three uses of zones in Sun Cluster:

R37.1: Logical nodes (1334 zones)

R37.2: Failover zones with HA Containers agent

R37.3: Zone Clusters

R38: [Phase 1] Global devices will be supported only via lofi. Use of /globaldevices partition will not be supported.

R38.1: [Phase 3] If the optional pxfs/global devices functionality is not present, global devices will not be supported at all.

R39: [Phase 1] Colorado will require that NWAM be disabled. The configuration tool will enforce this policy.

R40: [Phase 1] Cluster RBAC will be configured so that the useradministrator with the “Primary Administrator” rights profile can administer the cluster

R41: Colorado will not include support for RSMRDT (Remote Shared Memory Reliable Datagram Protocol).

2.2.7 Building and Development

R42: [Phase 3] Colorado will be buildable from source on OpenSolaris itself using a compiler available from the pkg.opensolaris.org package repository (eg. Sunstudioexpress).

R43: [Phases 1-3] Colorado will not be a fork of the Sun Cluster gate. The code will be integrated into the main development gate, using #ifdefs and make macros/flags where appropriate.

R44: [Phase 1] SCATE will run on Colorado on OpenSolaris.

2.2.8 Hardware Minimization

[Phase 1] Requirements captured as three sub-projects. See:

<http://opensolaris.org/os/project/colorado/Requirements/Colorado1-net.pdf>

<http://opensolaris.org/os/project/colorado/Requirements/iscsi-requirements-1.pdf>

<http://opensolaris.org/os/project/colorado/Requirements/colorado-haci.pdf>

R45: [Phase 2] Colorado will eliminate the requirement for shared storage on a two-node cluster by replacing local storage on each node making local disks highly available with AVS-SNDR. This is being run as a separate project. See <http://www.opensolaris.org/os/project/ohac-avs/>

R46: [Phase 3] Colorado will provide an arbitration mechanism for a two node cluster in which the cluster does not automatically takeover services if it can't distinguish between a node failure and a network partition. Human intervention is required to declare one of the nodes the “winner” in a split-brain case, or to tell the surviving node that it's ok to takeover in the node failure case.

R47: [Phase 3] Colorado will provide a plug-in model for arbitration mechanisms. Initial plug-ins will be the current “strong membership”, the “weak membership” from phase 1, and the

2.2.9 Software Minimization

R48: [Phase 3] PxFS and global devices shall be optional. See <http://opensolaris.org/os/project/colorado/Requirements/GDD-colorado.pdf>

R48.1: The memory footprint shall be reduced when pxfs is not present. See R1 in <http://opensolaris.org/os/project/colorado/Requirements/colorado-haci.pdf>

R49: [Phase 3] The RGM and other userland daemons' memory footprint shall be reduced. Specifically, the RGM shall not reserve huge amounts of swap or preallocate too many threads with large stack sizes.

R50: [Phase 3] Other optional components TBD.

2.2.10 Ease of Use

R51: [Phase 3] Colorado will include a BUI or GUI for ongoing administration similar to the current SCM. This BUI/GUI might need to be re-implemented to run on software available with OpenSolaris.

R52: [Phase 3] Colorado will include interfaces to support cluster in a software appliance. Specifically, these interfaces will allow the appliance to manage the cluster without exposing the cluster command-line or GUI/BUI to the end-user.

2.2.11 Documentation

R53: [Phase 1] Colorado will include simplified web-based introductory documentation similar to the OpenSolaris documentation available at <http://dlc.sun.com/osol/docs/content/IPS/getst1.html>

R53.1: The documentation will include a “conversion” of concepts and commands from popular competitors to OHAC

2.3 Exceptional Behavior

~~Same as in current product.~~

2.3.1 Errors Arising from User Interactions

Same as in current product: for existing interfaces. There will be a few new interfaces for managing the private interconnect over the public network, for managing the weak membership, and for managing the software modularity. These new interfaces will handle errors similarly to the current cluster interfaces.

The user could uninstall the cluster packages (using pkg uninstall) without first unconfiguring the cluster software, which would leave the machines in an unstable state. However, note that this is possible with the current SC 3.2 product as well.

Similarly, the user could uninstall optional packages without first disabling the functionality, which could leave the cluster in an unstable state.

The user could upgrade the underlying OS using pkg image-update to a build that the cluster software does not run on. This case will be handled by dependencies of the cluster packages on specific builds of the OS packages.

2.3.2 Errors Arising from System Failures

~~Same as in current product.~~ Because of the new hardware configurations allowed, there will be new error-handling for some failures. Specifically, with the weak membership implementation, a network partition will allow the cluster to remain operational in split-brain mode. Human intervention will be required for recovering from this situation. See the HACI requirements for details.

2.4 scchecks/eRAS checks

<TBD>

2.5 Reliability, Availability, Serviceability

Because of the relaxation of hardware requirements, reliability and availability can potentially be lower. Specifically, allowing network traffic on the public network and allowing a two-node cluster to run without a quorum device can lead to reduced reliability and availability.

2.6 Accessibility (Section 508) Requirements

Same as in current product.

2.7 Internationalization (I18N) and Localization (L10N) Requirements

Same as in current product.

2.8 Cluster Events

Same as in current product.

3. Efficiency Requirements

3.1 Time and Space Requirements (Performance)

~~This project is focused on reducing the hardware requirements and software footprint. It is not concerned with performance, although it is not expected to have significant performance impact.~~ However, performance must be considered.

It is expected that running the private interconnect over the public network may decrease performance of heartbeats and other transport. The performance must not be so degraded that lost heartbeats cause a cluster to declare a node to be dead when it is still alive.

It is also expected that the the weak membership model will decrease performance of failure detection and will add the performance hit of recovery from split-brain.

For production systems, we will recommend traditional hardware requirements for optimal performance.

4. References

[1] <http://www.opensolaris.com/>

[2] <http://opensolaris.org/os/project/pkg/>