1. What specifically is the proposal that we are reviewing?

- What is the technical content of the project?

This project adds support for NPIV (N Port ID Virtualization) to Solaris. NPIV is an addition to the FibreChannel specification to allow one physical fibrechannel port to simulate many ports.

This project also delivers enhancements for xVM and Zones to support NPIV. It is intended that an administrator assign one NPIV port per HBA to each client operating system instance. Logical Domains will be addressed in a future ARC case.

For further details, see functional specification, section 1.

- Is this a new product, or a change to a pre-existing one? If it is a change, would you consider it a "major", "minor", or "micro" change? See the Release Taxonomy in:

This is a change to the existing Solaris FibreChannel Stack, and supporting tools in xVM and Containers.

It will be released in a minor release.

- If your project is an evolution of a previous project, what changed from one version to another?

This project will add NPIV support to Leadville, the existing Solaris FibreChannel stack.

- What is the motivation for it, in general as well as specific terms? (Note that not everyone on the ARC will be an expert in the area.)

NPIV allows administrators to administer their FibreChannel SAN with virtual server using existing tools. Currently, with virtual servers, administrators can not use zoning/LUN masking because LUN masking is based on the one permanent WWN found in a FibreChannel Adapter. NPIV allows one WWN per adapter to be bound to the guest operating system.

- What are the expected benefits for Sun?

NPIV is a useful feature for system administrators. NPIV support for Solaris will keep Sun's virtualization strategy competitive. Microsoft virtualization, VMWare and Xen/Linux are all at different phases of supporting NPIV.

- By what criteria will you judge its success?

Deliver NPIV support to Solaris, including support for Xen and Containers.

2. Describe how your project changes the user experience, upon installation and during normal operation.

There is no required change for NPIV. However, users will be able to assign an NPIV port to each guest operating

system. With this NPIV port, system administrators will be able to user existing SAN management tools to segregate their physical SAN into logical zones. Zoning segregates visibility of FC nodes - only nodes in the zone have visibility to each other.

Additionally, administrators will be able to use NPIV to create virtual ports available for LUN masking. LUN masking is a function of SCSI target. LUN masking means that's for each target's lun, there is a list of initiator ports (virtual or physical) which can access that LUN.

- What does the user perceive when the system is upgraded from a previous release?

There is no default change in behavior.

3. What is its plan?

- What is its current status? Has a design review been done? Are there multiple delivery phases?

Current plan is for two deliveries. Design review was done in August.

We will deliver specific support for Logical domains in a later release, approximately Q4FY07. The enhancements we propose require the ability to dynamically add devices to a LDOM guest operating system instance. Logical Domains do not currently have this capability, but their team plans a Q4FY07 putback. We will have a separate PSARC case with details for Logical Domain Support.

4. Are there related projects in Sun?

- If so, what is the proposal's relationship to their work? Which not-yet- delivered Sun (or non-Sun) projects (libraries, hardware, etc.) does this project depend upon? What other projects, if any, depend on this one?

None. This project depends upon already released 4Gb Sun FibreChannel hardware.

- Are you updating, copying or changing functional areas maintained by other groups? How are you coordinating and communicating with them? Do they "approve" of what you propose? If not, please explain the areas of disagreement.

We are actively communicating with the Zones and Logical Domain team. They agree with our proposals.

We have started initial communication with container team.

5. How is the project delivered into the system?

- Identify packages, directories, libraries, databases, etc.

No new packages.
See functional spec for new files.

6. Describe the project's hardware platform dependencies.

Project requires new generation of hardware for NPIV
functionality. For Emulex and Qlogic, this is currently shipping 4Gb/S
hardware. Switches must also be modified to support NPIV;
switch manufacturers' 4Gb/S equipment supports NPIV.

Existing hardware will continue to function with existing
drivers, albeit without NPIV functionality.

- Explain any reasons why it would not work on both SPARC and Intel?

Both SPARC and Intel will be supported. Xen support will be
for Intel and LDOM for supported SPARC systems.

7. System administration

- How will the project's deliverables be installed and (re)configured?

Existing packages will be installed. They are all part of the
core Solaris install.

- How will the project's deliverables be uninstalled?

If desired, pkgrm can be used to remove the package

- Does it use inetd to start itself?
Not applicable

- Does it need installation within any global system tables?
No

- Does it use a naming service such as NIS, NIS+ or LDAP?
No

- What are its on-going maintenance requirements (e.g. Keeping global
tables up to date, trimming files)?
No

- How does this project's administrative mechanisms fit into Sun's system
administration strategies? E.g., how does it fit under the Solaris
Management Console (SMC) and Web-Based Enterprise Management (WBEM), how
does it make use of roles, authorizations and rights profiles?
Additionally, how does it provide for administrative audit in support of
the Solaris BSM configuration?

New command, fcadm, uses same roles as fcinfo(1)
Other commands are unchanged.

- What tunable parameters are exported? Can they be changed without
rebooting the system? Examples include, but are not limited to,
entries in /etc/system and ndd(8) parameters. What ranges are
appropriate for each tunable? What are the commitment levels

associated with each tunable (these are interfaces)?

Not applicable.

8. Reliability, Availability, Serviceability (RAS)

- Does the project make any material improvement to RAS?
No

- How can users/administrators diagnose failures or determine operational state? (For example, how could a user tell the difference between a failure and very slow performance?)

Main errors will be unsupported hardware. Diagnostic messages from fcadm will report unsupported hardware for example when trying to enable HBAs that don't support NPIV. Some errors will result in error message logs, for example when a switch is inserted to a SAN that does not support NPIV.

- What are the project's effects on boot time requirements?

Similar to adding new HBAs. Each NPIV port requires discovery similar to a physical HBA.

- How does the project handle dynamic reconfiguration (DR) events?

Existing support unchanged.

- What mechanisms are provided for continuous availability of service?

Not applicable

- Does the project call panic()? Explain why these panics cannot be avoided.

No.

- How are significant administrative or error conditions transmitted? SNMP traps? Email notification?

Error logging for asynchronous errors (for example plugging in an HBA to a switch which does not support NPIV).

- How does the project deal with failure and recovery?
- Does it ever require reboot? If so, explain why this situation cannot be avoided.

No, reboot is not required.

- How does your project deal with network failures (including partition and re- integration)? How do you handle the failure of hardware that your project depends on?

The driver responds similarly to existing FC behavior.

- Can it save/restore or checkpoint and recover?

No changes to existing behavior.

- Can its files be corrupted by failures? Does it clean up any locks/files after crashes?

    As with any file, the config can be corrupted by failure. In this case, it will be detected by checksum, and reinitialized. There are no temporary files to cleanup.

9. Observability

- Does the project export status, either via observable output (e.g., netstat) or via internal data structures (kstats)?

    Yes, current status is visible via fcinfo(1M)

- How would a user or administrator tell that this subsystem is or is not behaving as anticipated?

    Observe status from fcadm/fcinfo. Also observe error log.

- What statistics does the subsystem export, and by what mechanism?

    None

- What state information is logged?

    NPIV port enable/disable.

- In principle, would it be possible for a program to tune the activity of your project?

    Not applicable.

10. What are the security implications of this project?

- What security issues do you address in your project?
- The Solaris BSM configuration carries a Common Criteria (CC) Controlled Access Protection Profile (CAPP) -- Orange Book C2 -- and a Role Based Access Control Protection Profile (RBAC) -- rating, does the addition of your project effect this rating? E.g., does it introduce interfaces that make access or privilege decisions that are not audited, does it introduce removable media support that is not managed by the allocate subsystem, does it provide administration mechanisms that are not audited?
- Is system or subsystem security compromised in any way if your project's configuration files are corrupt or missing?
- Please justify the introduction of any (all) new setuid executables.
- Include a thorough description of the security assumptions, capabilities and any potential risks (possible attack points) being introduced by your project. A separate Security Questionnaire
    http://sac.sfbay/cgi-bin/bp.cgi?NAME=Security.bp
  is provided for more detailed guidance on the necessary information. Cases are encouraged to fill out and include the Security questionnaire (leveraging references to existing documentation) in the case materials.

   Projects must highlight information for the following important areas:
   - What features are newly visible on the network and how are they protected from exploitation (e.g. unauthorized access, eavesdropping)

- If the project makes decisions about which users, hosts, services, ... are allowed to access resources it manages, how is the requestor's identity determined and what data is used to determine if the access granted. Also how this data is protected from tampering.
- What privileges beyond what a common user (e.g. 'noaccess') can perform does this project require and why those are necessary.
- What parts of the project are active upon default install and how it can be turned off.

See section 2.7 of functional spec.

11. What is its UNIX operational environment:

- Which Solaris release(s) does it run on?

    Solaris Nevada

- Environment variables? Exit status? Signals issued?
  Signals caught? (See signal(3HEAD).)

    No changes

- Device drivers directly used (e.g. /dev/audio)?
  .rc/defaults or other resource/configuration files or databases?

    Uses /etc/cfg/fp/NPIV configuation file to persistently store WWN.

- Does it use any "hidden" (filename begins with ".") or temp files?

    No

- Does it use any locking files?

    No

- Command line or calling syntax:
  What options are supported? (please include man pages if available)
  Does it conform to getopt() parsing requirements?

    Yes. Conforms to getopt(). See man pages for fcadm, fcinfo

- Is there support for standard forms, e.g. "-display" for X programs?
  Are these propagated to sub-environments?

    Not applicable

- What shared libraries does it use? (Hint: if you have code use "ldd"
  and "dump -Lv")?

    fcadm will use
      libHBAAPI.so.1 =>   /usr/lib/libHBAAPI.so.1
      libc.so.1 =>   /lib/libc.so.1
      libcfgadm.so.1 =>   /usr/lib/libcfgadm.so.1
      libdevinfo.so.1 =>   /lib/libdevinfo.so.1
      libdl.so.1 =>   /lib/libdl.so.1
      libnvpair.so.1 =>   /lib/libnvpair.so.1
      libsec.so.1 =>   /lib/libsec.so.1
      libgen.so.1 =>   /lib/libgen.so.1

```
        libnsl.so.1 =>   /lib/libnsl.so.1
        libavl.so.1 =>   /lib/libavl.so.1
        libmp.so.2 =>    /lib/libmp.so.2
        libmd.so.1 =>    /lib/libmd.so.1
        libscf.so.1 =>   /lib/libscf.so.1
        libuutil.so.1 =>         /lib/libuutil.so.1
        libm.so.2 =>     /lib/libm.so.2
```

- Identify and justify the requirement for any static libraries.

      No static libraries

- Does it depend on kernel features not provided in your packages and not in the default kernel (e.g. Berkeley compatibility package, /usr/ccs, /usr/ucblib, optional kernel loadable modules)?

      None

- Is your project 64-bit clean/ready? If not, are there any architectural reasons why it would not work in a 64-bit environment? Does it interoperate with 64-bit versions?

      It is 64 clean/ready

- Does the project depend on particular versions of supporting software (especially Java virtual machines)? If so, do you deliver a private copy? What happens if a conflicting or incompatible version is already or subsequently installed on the system?

      None

- Is the project internationalized and localized?

      No.

- Is the project compatible with IPV6 interfaces and addresses?

      Not applicable

12. What is its window/desktop operational environment?

      Not applicable

13. What interfaces does your project import and export?

- Please provide a table of imported and exported interfaces, including stability levels. Pay close attention to the classification of these interfaces in the Interface Taxonomy -- e.g., "Committed," "Uncommitted," and "*Private;" see:

      http://sac.sfbay/cgi-bin/bp.cgi?NAME=interface_taxonomy.bp

  Use the following format:

See functional spec

Interfaces Imported		
Interface	Classification	Comments

```
The_referring_standard   Committed        ANSI Xy.Tz 1999 draft 37
Somebody_else ()   Consolidation Private


                  Interfaces Exported
Interface         Classification          Comments
My_subroutine_name      Committed
MY_MACRO          Project Private   <sys/project.h>
Etc, etc, etc...
```

- Exported public library APIs and ABIs
  Protocols (public or private)
  Drag and Drop
  ToolTalk
  Cut/Paste
- Other interfaces
- What other applications should it interoperate with? How will it do so?

n.a.

- Is it "pipeable"? How does it use stdin, stdout, stderr?

Yes, fcadm output will be pipeable.

- Explain the significant file formats, names, syntax, and semantics.

/etc/cfg/fp/NPIV will be XML, exact format undetermined

- Is there a public namespace? (Can third parties create names in your
  namespace?) How is this administered?

n.a.

- Are the externally visible interfaces documented clearly enough for a
  non-Sun client to use them successfully?

Yes. Documented with man pages, and updated documentation.

14. What are its other significant internal interfaces
    inter-subsystem and inter-invocation)?

- Protocols (public or private)
      HBA-API
      FibreChannel

- Private ToolTalk usage
      Not applicable
- Files
      /etc/cfg/fp/NPIV (note: private)

- Other
- Are the interfaces re-entrant?
      Existing restrictions apply. HBA-API will be re-entrant.

15. Is the interface extensible? How will the interface evolve?

- How is versioning handled?
      FCA and HBA API have embedded versioning as part of protocol.
      Leadville will fail to load on bad FCA versions.
      HBA API commands will fail

- What was the commitment level of the previous version?
      FCA: Project Private

- Can this version co-exist with existing standards and with earlier and later versions or with alternative implementations (perhaps by other vendors)?

    This standard is new.
    Existing FibreChannel HBAs will continue to work, with NPIV functionality.
    If used with NPIV capable HBAs with old switches, NPIV will not function.
    FC targets are all interoperable.

- What are the clients over which a change should be managed?
- How is transition to a new version to be accomplished? What are the consequences to ISV's and their customers?

    n.a.

16. How do the interfaces adapt to a changing world?

    Keep versioning in key interfaces, e.g. FCA and HBA API.

- What is its relationship with (or difficulties with) multimedia? 3D desktops? Nomadic computers? Storage-less clients? A networked file system model (i.e., a network-wide file manager)?

    Not applicable

17. Interoperability

- If applicable, explain your project's interoperability with the other major implementations in the industry. In particular, does it interoperate with Microsoft's implementation, if one exists?

    Compatible with other's implementations.

- What would be different about installing your project in a heterogeneous site instead of a homogeneous one (such as Sun)?

    None.

- Does your project assume that a Solaris-based system must be in control of the primary administrative node?

    No

18. Performance

- How will the project contribute (positively or negatively) to "system load" and "perceived performance"?

    In general, unchanged. Some events, such as RSCN changes, generated when an event on the SAN occurs, will take additional processing compared to not using NPIV (because a separate event will be generated for each NPIV WWN). Again, it should be similar to adding another physical HBA without NPIV.

- What are the performance goals of the project? How were they evaluated? What is the test or reference platform?

    Unchanged performance, except in the RSCN case. Again, processing for SAN events (State Change Notifications), will be similar to multiple HBAs on the SAN.

- Does the application pause for significant amounts of time?
  Can the user interact with the application while it is performing
  long-duration tasks?

      No, there should be no pauses.

- What is your project's MT model? How does it use threads internally?
  How does it expect its client to use threads? If it uses callbacks,
  can the called entity create a thread and recursively call back?

      No changes to threading model. HBA API library, by
      specification, supports multi threading, however, our
        application is single threaded.

- What is the impact on overall system performance? What is the
  average working set of this component? How much of this is
  shared/sharable by other apps?

      Unchanged system performance. Again, adding a virtual port is
      analogous to adding a separate physical HBA both in processing
      an amount of memory required for data structures.

- Does this application "wake up" periodically? How often and under
  what conditions? What is the working set associated with this
        behavior?

      No, there is no polling

- Will it require large files/databases (for example, new fonts)?

      No. Database should be small, approximately 100 bytes per WWN.

- Do files, databases or heap space tend to grow with time/load? What
  mechanisms does the user have to use to control this? What happens to
  performance/system load?

      No, memory is dynamically allocated when new NPIV WWN are
      created, and freed when the WWN is deleted.

19. Please identify any issues that you would like the ARC to address.

- Interface classification, deviations from standards, architectural
  conflicts, release constraints...
- Are there issues or related projects that the ARC should advise the
  appropriate steering committees?

  None.

20. Appendices to include

- One-Pager.
- Prototype specification.
- References to other documents. (Place copies in case directory.)