# Solaris Open Fabrics User Verbs Architecture Document

| | |
|---|---|
| Editor | Pramod Gunjikar |
| Authors | Pramod Gunjikar<br>Brendan Doyle |
| Date | 09/02/09 |
| Version | 0.6 |
| Last Modified by | brendan.doyle@sun.com |
| File name | solaris_ofuv_arch |

# Table of Contents

# 1 Revision History

| *Revision* | *Date* | *Summary* |
|---|---|---|
| 0.1 | 03/16/07 | Initial Draft Version. |
| 0.2 | 03/22/07 | After internal review |
| 0.3 | 07/11/07 | Revised Phase-I Architecture |
| 0.4 | 07/11/07 | Updated Arch, and redefined phase I&II |
| 0.5 | 02/05/09 | Update with re-architecture for in kernel RDMA CM and add OpenSM Architecture |
| 0.6 | 08/28/09 | Update with re-architecture for in kernel verbs. |

## 1.1 References

*OpenIB Architectural Overview – Roland Dreier and others*

*OpenFabrics Software Stack – June 2006*

*OFED Update – Presentation in IBTA-OFA presentation on Sept. 2006*

*UDAPL Porting Guide, version 8.11*

*IBTF Architecture Document*

*Solaris Open Fabrics User Verbs Implementation Details - July 30, 2008*

*Open MPI code base at : http://www.open-mpi.org/*

*Open Fabrics User Verbs (OFUV) Primary Kernel Components Test Plan - Sept 2009*

http://ontestreview.central.sun.com/wiki/index.php/Open_Fabrics_User_verbs_(OFUV)_Primary_Kernel_Components

# 2 Glossary

| | |
|---|---|
| **OFED** | Open Fabrics Enterprise Distribution |
| **OFUV** | Open Fabrics User Verbs |
| **ULP** | Upper Layer Protocol |
| **BTL** | Byte Transport Layer |
| **uDAPL** | User Direct Access Programming Library |
| **MAD** | Management Datagram |
| **DR** | Dynamic Re-configuration |
| **IBTF** | Infiniband Transport Framework |
| **IBTF VTI** | IBTF Verbs Transport Interface |
| **DAL** | Datagram Acceleration Layer |
| **MCE** | Multicast Extension Library |

# 3 Introduction

The OpenFabrics Enterprise Distribution (OFED) supports InfiniBand and iWARP fabrics. OFED is developed by the OpenFabrics Alliance (OFA) See www.openfabrics.org, it is a collaborative open source development. OFED is currently only supported on Linux environments. The OFED distribution consists of multiple components. The OpenFabrics User Verbs (OFUV) API is a key component of OFED. The user verbs API provides userlevel access and kernel by pass to the Infiniband (IB) transport for:

- ULPs; like OpenMPI, OpenSM, etc..
- Middle ware; like Voltaire's Multicast Extensions (MCE),
- RNA Networks, & others.
- Typically providing verbs API extensions to financial services
- software for companies such as Wombat, Reuters, etc.
- Diagnostics utilities and tools.

This project will provide the same OFED user verbs APIs on Solaris, such that no, or minimal, changes will be required to port Linux OFED verbs applications to Solaris.

The Open-MPI source code uses OFUV interfaces for IB transport functionality. The current Solaris Open MPI has been implemented using a User Direct Access Programming Library (uDAPL) Byte Transport Layer (BTL), as Solaris OFUV support is not currently available. There is a large community effort developing Open MPI using the Open Fabrics User Level Verbs API. The Solaris Open MPI implementation can leverage these community efforts better, by adopting to Solaris OFUV implementation in the future. The Open MPI implementation using OFUV is expected to be more efficient than the implementation using uDAPL.

Voltaire (Multicast Extensions (MCE) library (port), Cisco (Datagram Acceleration Layer (DAL) port) **,** Lustre and Open MPI are the initial potential consumers of Solaris OFUV interfaces. Subnet Manager related diagnostic tools are also planning to use OFUV interfaces in the future. Solaris OFUV support will also be an enabler for porting ULPs and tools available within OFED and in other distributions, to Solaris.

This document defines the OFUV architecture and details the functional, development and testing requirements of OFUV support on Solaris.

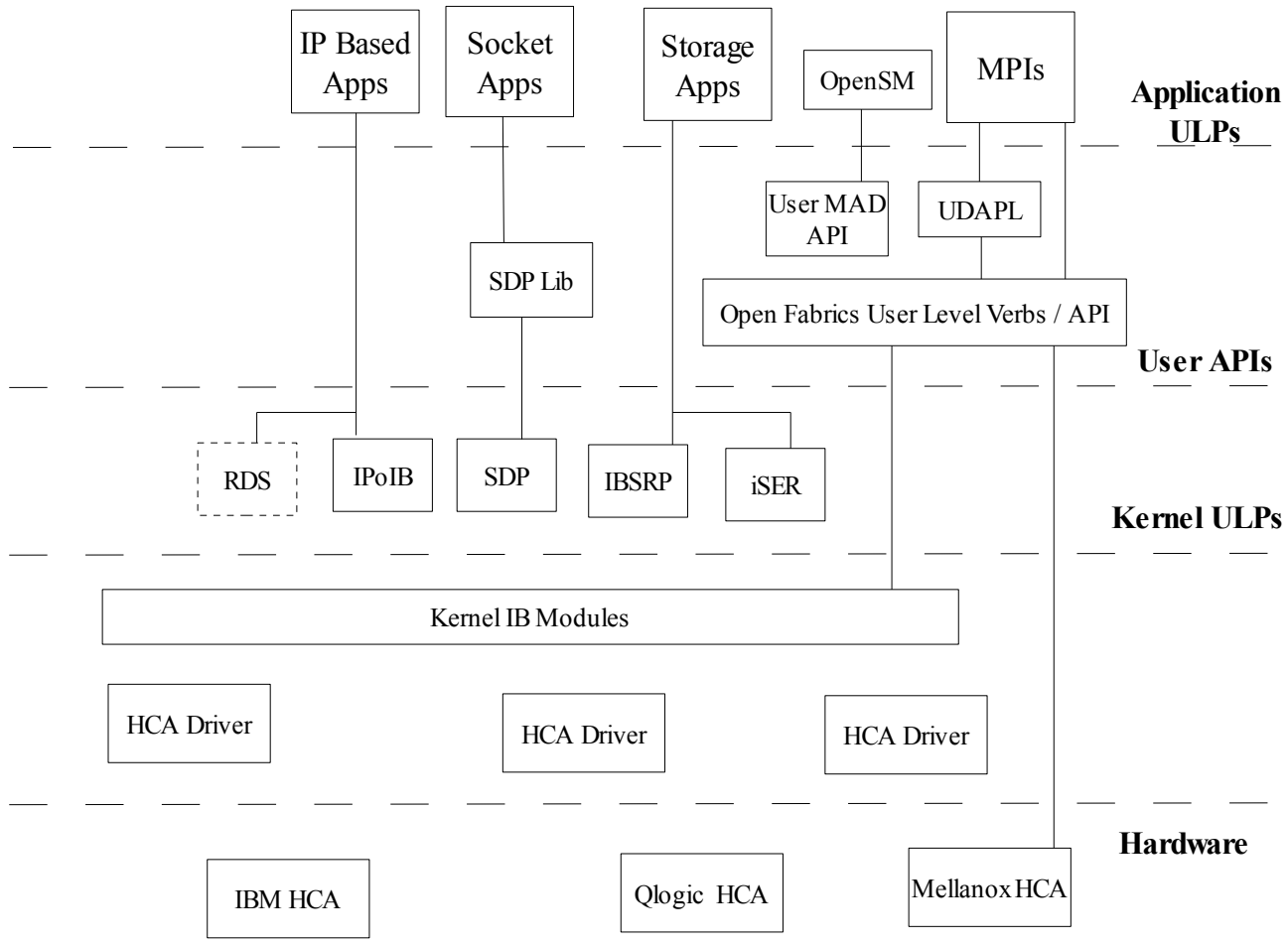
# 4 Open Fabrics Enterprise Distribution Architecture

## 4.1 Components

The OFED Linux implementation consists of multiple components:

1. Various MPIs - Open MPI, MPICH
2. Upper Layer Protocols – SRP, SDP, RDS , iSER, uDAPL
3. Open Fabrics User Level Verbs / API (OFUV)
4. User Level MAD API
5. User level SDP library
6. Open SM
7. Test and Performance Tools

The general architecture of OFED stack is as represented in the diagram below.



## 4.2 OF User Level Verbs API

The OF User Level Verbs API provides support to Application level ULPs like MPI and Lustre. The OFUV interfaces provide support for Infiniband verbs (RC, UD & UC transport services are supported). Additionally libraries for supporting communication management, MAD transport and SA Access are also provided. The components of OF User Level library interfaces, that are of interest to this project are :

1. Verbs library - *libibverbs*
2. User level HCA Drivers
    - User level Drivers for Mellanox HCAs – *libmthca & libmlx4*
1. Communication Management library
    - *librdmacm*
1. MAD transport library and SA Access library – *libibmad & libibumad*
2. Support for transport services : RC/UD

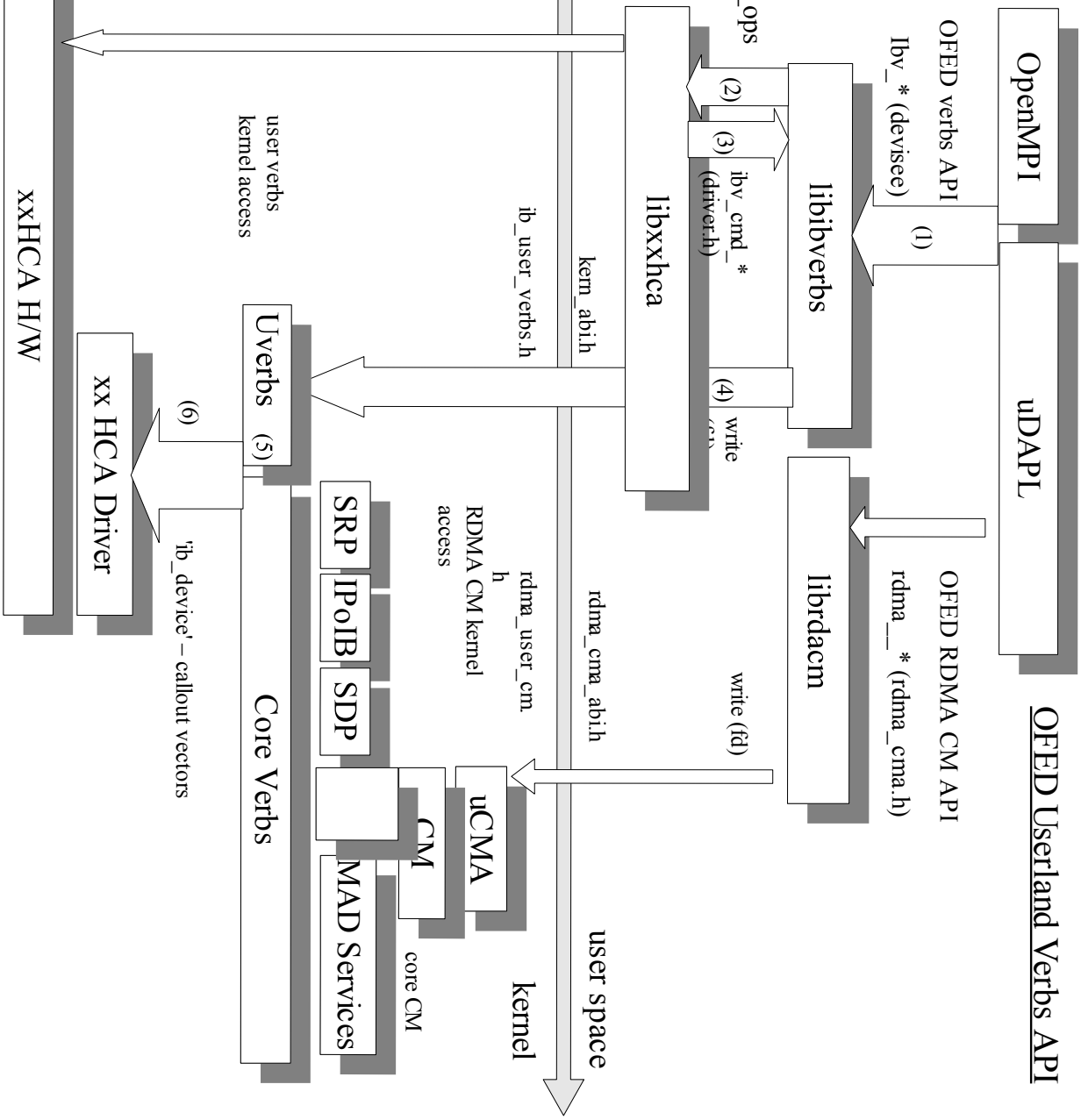
The Verbs library provides an API which maps to the IB Verbs, as specified in Volume 1 (not all verbs are supported, only those that are version 1.1 compliant are supported). Application level ULPs like Open MPI and Lustre use the Verbs library for communicating over IB. The user level HCA drivers are plug-in libraries which provide Verbs support for a class of HCAs. The OFED libraries on Linux provide support for Mellanox, Pathscale/QLogic and IBM HCAs. This project will support Mellanox HCAs only. All other HCAs are out of scope of this project.

The *librdmacm* library provides connection management functionality. This library provides a generic RDMA set of CM interfaces that can run over iWARP and IB. *librdmacm* also provides abstracted SA access functionality in "find path" APIs. The UDAPL implementation in OFED uses the *librdmacm* library functionality. The *libibcm* library also provides API that support for ULPs for connection establishment. The *librdmacm* is expected to replace *libibcm*. Support for *libibcm* is not of interest for the initial phases of this project.

The SA access library provides interfaces for querying IB subnet information. MPICH implementations is the only current user of the SA access functionality. Currently the only SA access functionality which is supported by OFED 1.0 is the ability to retrieve PathRecords and manage multicast groups.

OFED user land libraries support transport services for: Reliable Connected(RC), Unreliable Datagram (UD) and Unreliable Connect (UC).

The diagram overleaf represents the OFUV architecture and interfaces on Linux.


OFED Userland Verbs API
OpenMPI
uDAPL
OFED verbs API
Ibv_ * (devisee)
(1)
OFED RDMA CM API
rdma_cma.h)
rdma_ * (rdma_
libibverbs
librdacm
ibv_context_ops
(verbs.h)
(2)
(3)
ibv_cmd_ *
(driver.h)
XX HCA plug-in
library
libxxhca
write
write (fd)
Door Bell
kern_abi.h
ib_user_verbs.h
rdma_cma_abi.h
rdma_user_cm
h
RDMA CM kernel
access
user verbs
kernel access
Uverbs (5)
(6)
SRP
IPoIB
SDP
uCMA
CM
MAD Services
core CM
kernel
user space
xx HCA Driver
'ib_device' – callout vectors
Core Verbs
xxHCA H/W

The API flow is:

1. Application makes ibv_* API call into libibverbs
2. libibverbs calls into HCA library via ibv_context_ops vector
3. HCA library performs HCA specific operations and calls back into libibverbs via ibv_cmd_* API
4. libibverbs sends kern_abi.h cmd to uVerbs kernel access module via write on fd
5. uVerbs calls HCA userland specific entry point via ib_device vector
6. HCA userland entry point performs userland specific operations and calls core HCA entry point

Create QP example:

1. Application calls ibv_create_qp()
2. ibv_create_qp() calls mthca_create_qp() via ibv_context_ops.create_qp() vector
3. MT HCA library performs MT specific operations and calls back into libibverbs via ibv_cmd_create_qp()
4. libibverbs calls into uVerbs kernel entry point ib_uverbs_create_qp() via kern_abi.h cmd write on fd
5. ib_uverbs_create_qp() calls mthca_create_qp() via ib_device.create_qp() vector
6. mthca_create_qp() performs userland specific operations and calls mthca_alloc_qp()

# 4 Solaris OFUV Overview

## 4.1 Solaris OFUV Goals and Non Goals

### 4.1.1 Goals

1. Support all OFUV APIs exposed to ULPs, conforming to OFED header files. Linux specific components (like sysfs related parameters) may either be emulated or have Solaris equivalent structures

2. Support for Open Fabrics Subnet Access and User MAD libraries. Initially just the OFED *librdmacm* library which provides interfaces for SA PathRecord lookups. Later Phases will add *libibmad* & *libibumad* support which will enable the port of Open SM and other MAD based utilities at a later date.

3. No degradation of latency and bandwidth with respect to Linux OFED implementation. For example, the number of copies during I/O has to be equal to or less than the OFED Linux implementation.

4. Add Solaris specific enhancements for supporting features such as , DR, etc.

5. Optimal use of the existing Solaris IB code.

6. Seamless integration of other Open Fabrics APIs from Linux.

7. The Solaris OFUV architecture should be open to accommodate other IB HCAs like Pathscale and iWARP controllers.

8. The performance of Solaris OFUV will be equal to or better than that of Solaris UDAPL implementation. The performance of Open MPI using UDAPL BTL and OFUV BTL will be the basis for performance comparison.

9. This project will support IB verbs, as specified in InfiniBand Architecture Specification Volume 1, Release 1.1, as supported by OFED 1.3.

10. Kernel components to support Open fabrics kernel verbs KPI and RDMA CM KPI.

### 4.1.2 Non-Goals

1. No plan for supporting *libibcm*. *libibcm* is planned to be depreciated.

## 4.2 Solaris OFUV Project Scope

Solaris OFUV includes all interfaces required for Application level IB ULPs. Solaris specific requirements like DR support, will also be added. This project will be will be rolled out in a number of phases. The initial phase will provide interfaces required for supporting Voltaire MCE and Open MPI.

Later Phases will add support for OpenSM and other MAD based OFED diagnostic tools and utilities, phase III will add add support for any remaining ULPS and OFED tools/utilities. The details are given below:

| ***OFUV Component*** | ***Solaris Support*** | ***Comments*** |
|---|---|---|
| libibverbs | Initial Phase | Required by MCE, Open MPI and Lustre |
| libmthca | Initial Phase | Required by MCE, Open MPI and Lustre |
| libmlx4hca | Initial Phase | Support of Hermon is hard requirement for MCE |
| RC | Initial Phase | Required by Open MPI and Lustre. ULPs can use Out Of Band protocols (as in Open MPI) or *librdmacm* to obtain the remote information required for establishing RC connection. |
| librdmacm | Initial Phase | Required by MCE and Lustre |
| UD | Phase I | MCE based on UD |
| ibibumad &libibmad | Later Phase | Diagnostic tools, SM utilities use the MAD library. Will be layered on top of IBMF |
| *libibpathverbs*, *libehca* and other HCAs | Support when Solaris supports additional HCAs. | Support for OFUV user plug-in library can be part of the overall support for a new HCA on Solaris. |
| SA access libraries | It is not known if this support will be required. Any project to support SA access libraries on Solaris should be independent of this project. | SA access library (osmv_query_sa() and related calls) are used by MPICH and some diagnostic tools. There is no requirement for MPICH support on Solaris.<br><br>It is not known if this support will be required in the future |
| UC | The requirement for UC at user level, can be re-evaluated in the future. | Solaris IB kernel infrastructure currently does not support Unreliable connect on IB fabric. There is currently no known requirement for UC support in the user level. This can be re-evaluated in the future. |
| *libibcm* | Not planned at this stage | OFED is migrating from libibcm to librdmacm for providing connection management functionality for ULPs. |

## 4.3 Risks and Assumptions

1. Any requirement for support for ULPs other than the identified can change the requirements.
2. No major changes are expected in the existing Solaris IB software stack for. Any requirement

for major change will affect this project.

# 5 Solaris OFUV Architecture

## 5.1 Architecture

The architecture/implementation is based on porting OFED source for *libibverbs, libmthca, libmlx4, librdmacm, libibmad & libibumad*, and creating four new kernel modules (*sol_uverbs*, *sol_ucma*, *sol_ofs* and *sol_umad*) that provide the expected OFED userland/kernel interface to the ported OFED libraries, and then interface to the Solaris kernel via the InfiniBand Transport Framework (IBTF). The kernel modules will be part of the ON consolidation, however the ported OFED userland code will either be part of the Sun FreeWare (SFW) consolidation or put back to the OFA community. This dual consolidation split is necessary in order to maintain the project goal of porting OFED code to Solaris with minimal or no changes, so that the task of updating to a new OFED version is also minimal. OFED code does not conform to ON cstyle standards, and relies on GNU automake tools to be built, thus to incorporate it into ON would require major changes, and this would make the task of keeping up to date with newer versions of OFED very difficult The Solaris OFUV implementation phase-I will support interfaces defined in the OFED *verbs.h* & *rdma_cm.h* headers, and exported by the OFED *libibverbs* & *librdmacm* libraries (phases II & III will add support for additional libraries and applications). Linux specific components (like sysfs related parameters) will be emulated.

HCA User Level plug-ins use the following interfaces to interact with the kernel

- Interface with a Solaris kernel agent (*sol_uverbs*) supporting OFED verbs API
- Interface to HCA hardware for Kernel bypass/fastpath

The interfaces defined in *dapl_tavor_hw.h*/*ib_mlnx_umap.h* provide the kernel bypass mechanism for the uDAPL tavor plug-in The tavor(7d) & *hermon(7d)* drivers contains the following enhancements to support OS-bypass for uDAPL:

- mmapping the UAR doorbell page for synchronizing with the HCA hardware
- mmaping CQs for OS bypass polling of work request completions
- mmaping QPs for OS bypass posting of work requests
- Locking down of user memory during MR registration.

The ported OFED *libmthca* & *libmlx4* libraries will use this model for mapping Doorbells, CQE & WQE memory required for Kernel bypass/fastpath.

The interface with the Solaris kernel agent supporting OFED, should support all verbs, which includes support for non-protected verbs (for HCAs which do not support kernel bypass for non-protected mode verbs). The interface with the kernel agent for OFED is based on OFED interfaces as defined in *kern_abi.h* and *ib_user_verbs.h*.

Two later requirements were added to the kernel architecture of this project, 1) The architecture should support the provision of an OpenFabrics kernel RDMA CM API (Lustre and RDSv3) and 2) The Architecture should support the provision of an OpenFabrics kernel verbs API (RDSv3), these will be are provided by the *'sol_ofs'* kernel module. The kernel components will be delivered in two phases as a result of these two late requirements. Phase-1 is derived from kernel architecture implemented prior to these requirements, with 'sol_ofs' being extended to add the kernel API support. Phase-2 will involves a modification to the kernel architecture with 'sol_uvebrs' being modified to be a client of 'sol_ofs'. 'sol_umad' is also added in phase-2 to provide support for OpenSM, and OpenFabrics MAD based tools/utilities and libraries.

The figure overleaf gives an overview of the Phase-1 architecture, with notes as follows:

1. Based on port of OFED libxxhca, with SOLARIS specific code. Uses CI data IN/OUT model and mmap() calls to map memory for WQE and CQE buffers.

2. Uses OFED doorbell calls.

3. New Kernel Module, Upper interface OFED lower interface IBTF.

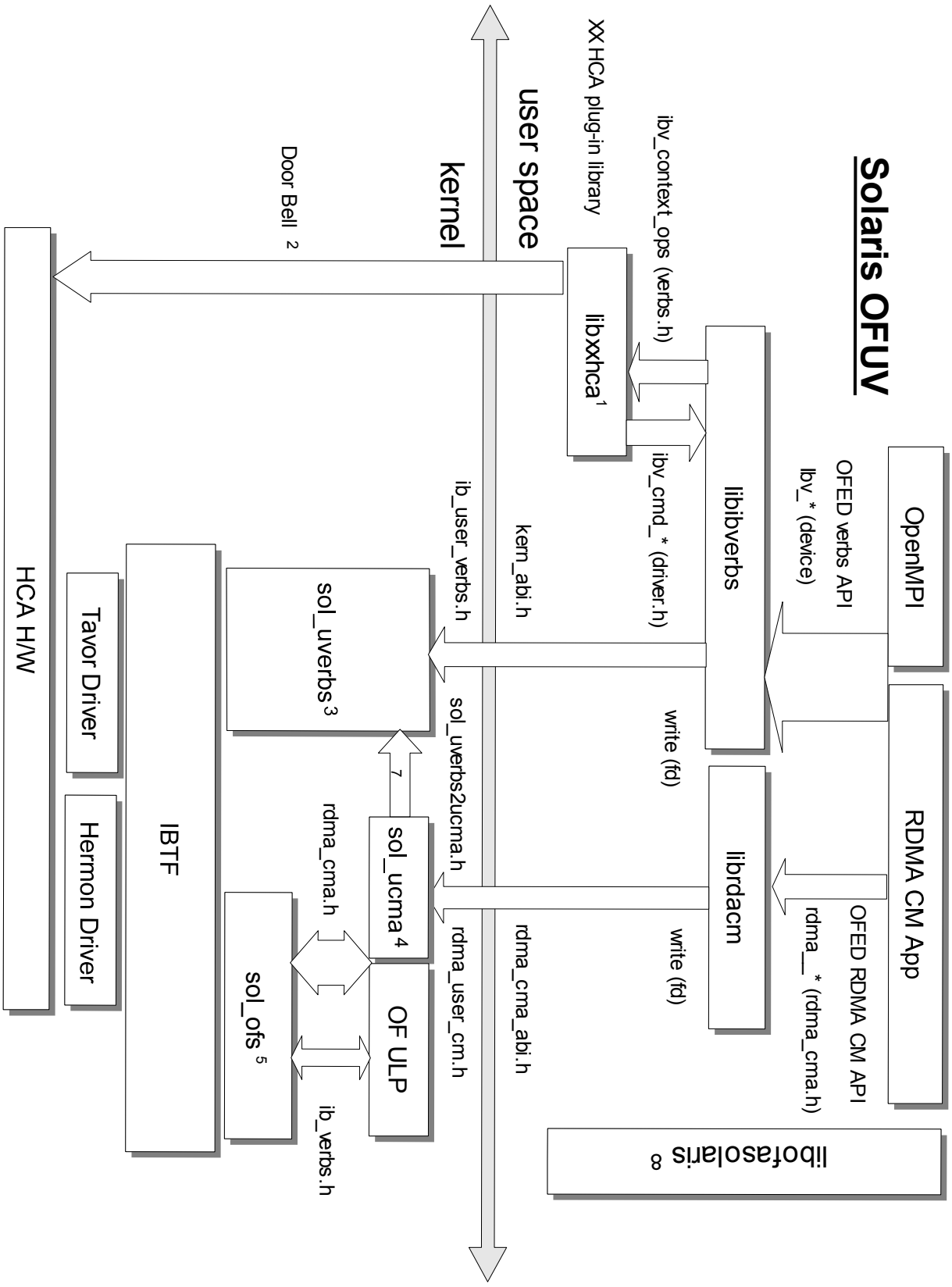
4. New kernel Module implements OFED RDMA ABI into userspace (rdma_user_cm.h).

5. New kernel module implements Transport agnostic (iwarp and IB) interfaces defined by rdma_cm.h, and the Open Fabrics kernel verbs API defined by ib_verbs.h

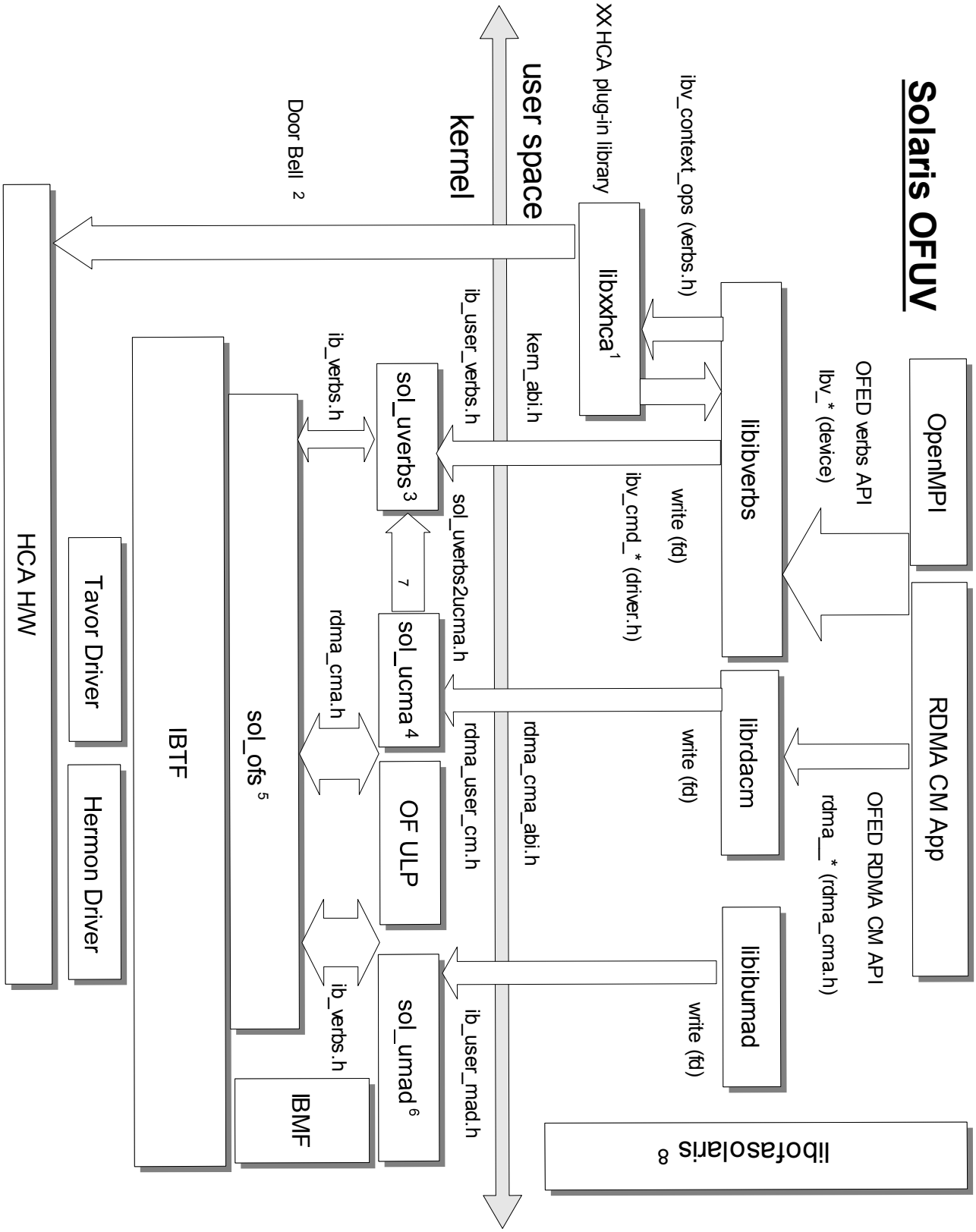
6. New kernel misc module that implements interfaces OpenSM into IBMF

7. Private Interface to obtain IBTF handles from librdma identifiers.

8. libofasolaris.so new library that provides "Linux like" or "missing" capability not directly available on Solaris. linked with the user space libraries and applications.

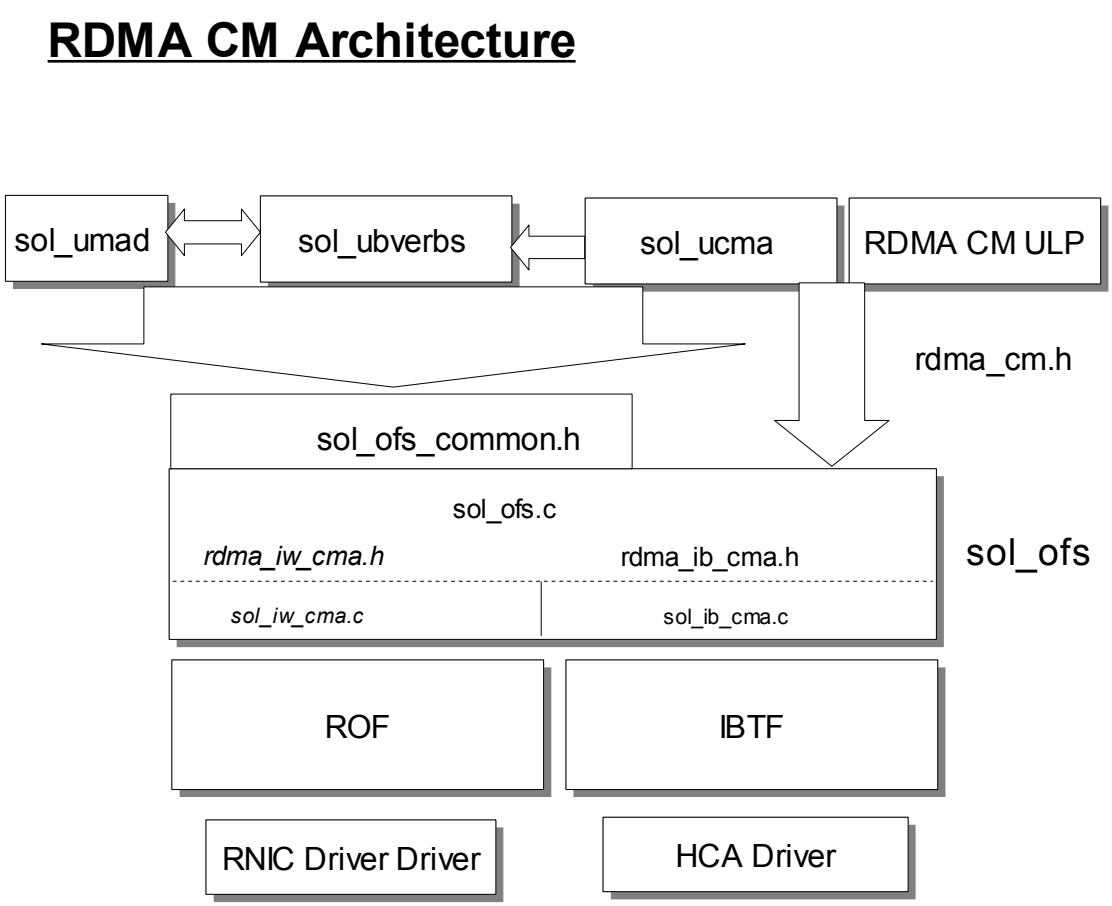

Solaris OFUV
OpenMPI
RDMA CM App
OFED verbs API
Ibv_* (device)
OFED RDMA CM API
rdma_*_* (rdma_cma.h)
libibverbs
librdacm
libofasolaris 8
ibv_context_ops (verbs.h)
ibv_cmd_* (driver.h)
write (fd)
write (fd)
XX HCA plug-in library
libxxhca 1
user space
kernel
Door Bell 2
ib_user_verbs.h
kern_abi.h
sol_uverbs2ucma.h
rdma_cma_abi.h
rdma_user_cm.h
sol_uverbs 3
sol_ucma 4
OF ULP
rdma_cma.h
sol_ofs 5
ib_verbs.h
IBTF
Tavor Driver
Hermon Driver
HCA H/W

The figure overleaf gives an overview of the Phase-2 architecture, here '*sol_uverbs'* is made a client of '*sol_ofs'* and *'sol_mad'* is also added as a client of '*sol_ofs'*.


Solaris OFUV
OpenMPI
RDMA CM App
OFED verbs API
ibv_* (device)
OFED RDMA CM API
rdma_* (rdma_cma.h)
libibverbs
librdacm
libibumad
libofasolaris 8
write (fd)
write (fd)
write (fd)
ibv_context_ops (verbs.h)
XX HCA plug-in library
libxxhca 1
ibv_cmd_* (driver.h)
user space
kernel
Door Bell 2
ib_user_verbs.h
kern_abi.h
rdma_cma_abi.h
rdma_user_cm.h
ib_user_mad.h
ib_verbs.h
sol_uverbs 3
sol_uverbs2ucma.h
7
sol_ucma 4
rdma_cma.h
OF ULP
sol_umad 6
ib_verbs.h
sol_ofs 5
IBMF
IBTF
Tavor Driver
Hermon Driver
HCA H/W

## 5.2 Solaris RDMA CM architecture

The Solaris OFUV *librdmacm* will provide interfaces as defined in the OFED *rdma_cma.h* header to application ULPs. The *librdmacm* library will interface with the a separate *sol_ucma* kernel module using interfaces as defined by the OFED *rdma_cma_abi.h* header. The *sol_ucma* kernel module implements the OFED interfaces defined by *rdma_cma_abi.h* and *rdma_usr_cm.h* calling into the the in kernel '*sol_ofs'* module using the OFED defined *rdma_cm.h* interfaces. This allows kernel ULPs to also use the OFED RDMA CM APIs (a requirement from Lustre). The *sol_ofs* module then implements transport specific modules operations, calling into either IBTF for IB or ROF for iWARP. The diagram below illustrates the RDMA CM architecture.




The "*sol_ofs*" module contains a generic *sol_cma.c* and transport specific *sol_ib_cma.c* and *sol_iw_cma.c* (future) source files. The interface between the generic and the transport specific portion of *sol_cma*, is the header file <*rdma_ib_cma.h*> and <*rdma_iw_cma.h*> (future). The transport specific header file <*rdma_xy_cm.h*> will contain all interface functions defined in Solaris *rdma_cm.h* with following changes :

1. No transport specific rdma_create_id().
2. All API function names have rdma_ib_ / rdma_iw_ prefix.

The *sol_uverbs* module exports the following functions (as defined by the sol_ucma2uverbs.h) to *sol_ucma*:

*sol_uverbs_get_clnt_hdl();*

*sol_uverbs_qpnum2uqpid();*

*sol_uverbs_uqpid2qphdl();*

*sol_uverbs_disable_uqp_modify();*

These functions enable *sol_ucma* to obtain mappings from userland handles to kernel handles. The *sol_uverbs_disable_uqp_modify()* is required to disable direct modification of QP state by librdmacm, in IBTF this is done by the Solaris IBCM.

The *sol_ofs_common.h* header defines the following common OFUV utility functions:

1. APIs for generic linked list management
2. APIs for sol_ofs_dprintf_l*() debug routines

All OFUV related modules can use these interfaces.

*'sol_uverbs'* maintains a list of HCA devices and ports, It creates a character minor device node for each HCA reported by the IBTF. These device nodes are utilized by *libibverbs* to open specific HCA instances. *'sol_uverbs'* exports device properties for each minor device that may be used to assist in determining the proper character minor device to open (See Solaris Open Fabrics User Verbs Implementation Details - July 30, 2008 for more details).

## 5.2 *OFUV Library Path*

All libraries will be located in the */usr/lib/* directory. The typedef for the userland initialization function pointer *ibv_driver_init_func*() defined in *driver.h* will be used as in Linux OFED.

## 5.3 Loading of Userland Drivers and kernel modules

The *libibverbs* library will load userland drivers from the */usr/lib/* directory, by default. The library will also use the path specified by the calling user's environment, using the environment variable, *OPENFABRICS_DRIVER_PATH*_ENV. This is similar to the OFED implementation on Linux.

All drivers pre-load the *sol_ofs* using : ld -N misc/sol_ofs The drivers are : *sol_ucma*, *sol_uverbs* & *sol_umad*. *sol_ucma* loads *sol_uverbs* (if *sol_uverbs* has not been loaded yet) using ddi_modload() and friends. An RDMA CM ULP kernel client will preload *sol_ofs*, using ld -N misc/sol_ofs

## 5.4 Solaris extensions to libibverbs

### 5.4.1 Discovery of HW providing Verbs functionality

The kernel agent providing the Verbs functionality extends the OFED user/kernel interface defined in kern-abi.h and ib_user_verbs.h to add a number of SOLARIS specific commands which will return the information related to all HW controllers providing Verbs functionality.

### 5.4.2 Dynamic Reconfiguration (DR)

OFED does not completely support Solaris specific features like HCA DR and memory DR . The Solaris OFUV project will define methods for ULPs to support these Solaris specific features. These methods can be based on the existing Solaris support like RCM (Re-Configuration Manager) or extensions to the OFUV interfaces. The details of the Solaris specific extensions have not been finalized.

The Solaris specific extensions will be designed in such a manner, that it is independent of the OFUV interfaces. The ULPs should be able to support the Solaris specific extensions with zero or very minimal changes in the ULP code using the OFUV interfaces (defined in *verbs.h*). The Solaris specific extensions will be defined in separate header file(s).

For the Open MPI/Lustre implementation, the intent is to just add source code to support the Solaris specific extensions, with no or very little modifications to the Open MPI BTL/Lustre sources. Details of achieving the same have to be finalized.

OFUV kernel modules handle HCA DR as follows:

1. *'sol_ofs'* is the only IBTF client, it receives all IBTF HCA DR async events.
2. For HCA_DETACH events *'sol_ofs'* calls the “remove” callback for each of it's clients.
3. The “remove” callback in '*sol_uverbs'* generates a new IB_EVENT_HCA_DETACH (Solaris

specific extension) and queues it to userland clients. Userland clients should free resources for this HCA and return SUCCESS. Freeing of resources, includes destroying CM IDs using the HCA.

## 5.6 Solaris IBTF/HCA driver modifications

The Solaris IB Transport framework and HCA drivers may require minor modifications for supporting OFUV. The intent is to keep the changes minimal. However IBTF changes may be required for functionality or for better performance of OF Verbs.

# 6 Additional Phases

Later phases adds the following:

- Provide support for OFED MAD libraries – *libibmad* & *libibunad*
- XRC
- Port OpenSM to Solaris
- DR Support.
- test/perf/management/other applications of OFED that have not already been ported,