Technical Overview of iSCSI Extensions for RDMA (iSER) & Datamover Architecture for iSCSI (DA)

RDMA Consortium

Mike Ko

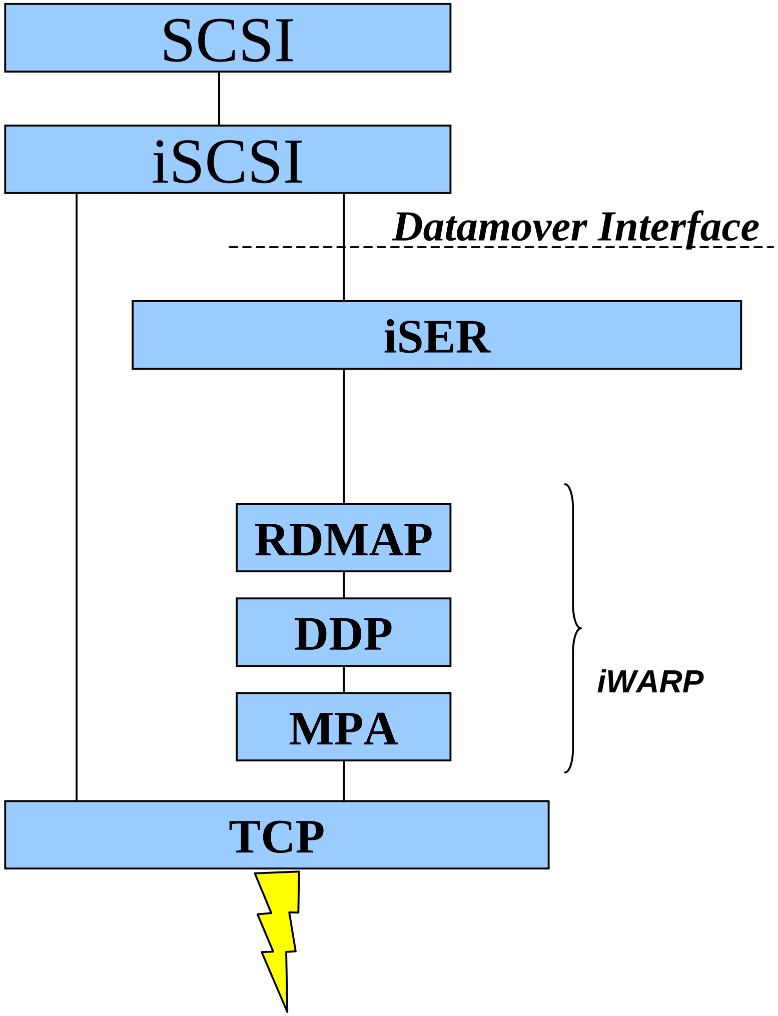
July 21, 2003

Outline

- Introduction
- iSCSI Control-type vs. Data-type PDUs
- Operational Primitives
- iSER Header
- SCSI Write Operation
- SCSI Read Operation
- Flow Control
- Connection Setup for iSER-assisted Mode
- Connection Termination for iSER-assisted Mode
- Error Recovery
- Summary

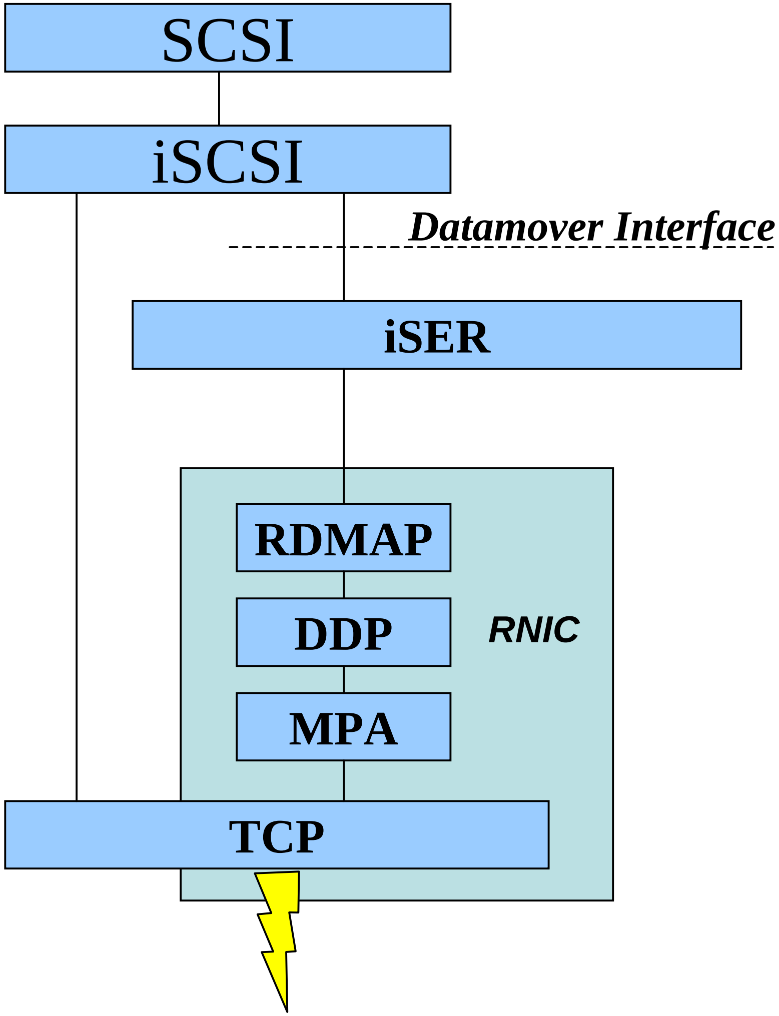
What are DA and iSER?

- The Datamover Architecture defines an abstract model in which the movement of data between iSCSI end nodes is logically separated from the rest of the iSCSI protocol
  - Allows a datamover protocol layer to offload the tasks of data movement and placement from the iSCSI layer
- The iSCSI Extensions for RDMA (iSER) protocol is one such datamover protocol
  - Applies the Datamover Architecture in extending the data transfer capabilities of iSCSI to include RDMA (Remote Direct Memory Access) as defined in the iWARP protocol suite
    - The iWARP protocol suite, submitted by the RDMA Consortium to the IETF for standardization consideration in October 2002, provides the RDMAP and DDP (Direct Data Placement) functionality to the IP fabric

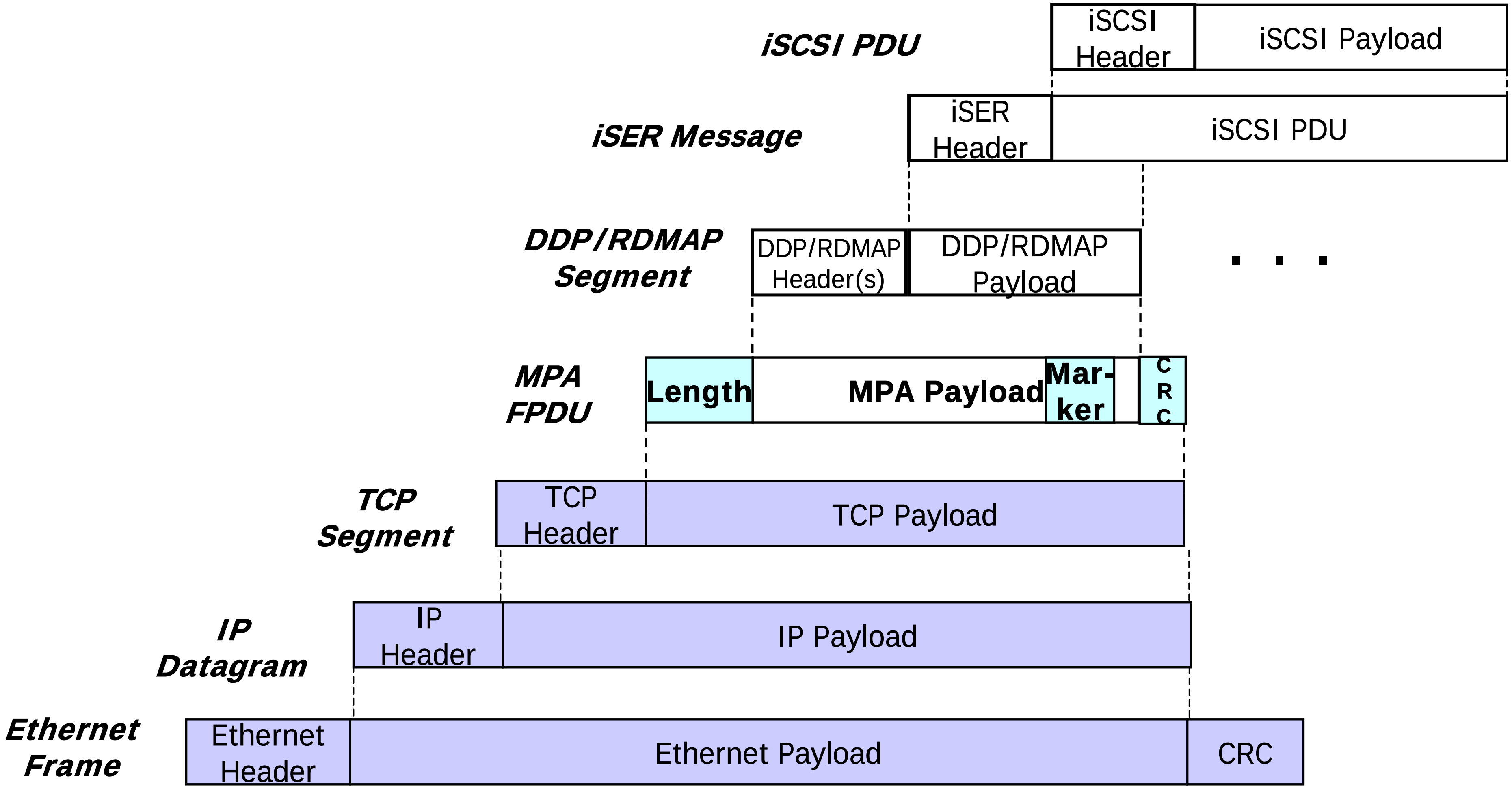


iSER & RNICs

- The iSCSI Extensions for RDMA (iSER) protocol
  - Takes advantage of the generic direct data placement mechanism and RDMA semantics offered by the iWARP technology instead of being iSCSI specific
  - Allows iSCSI implementations to have data transfers which achieve true zero copy behavior using generic RDMA network interface controllers (**RNIC**s)
    - True zero copy eliminates the increasing memory-to-memory copy overhead incurred in network protocol processing, particularly in the receive path, as speeds grow to 10 Gb/s and beyond



Example of an Encapsulation of an iSER Payload in an iWARP Message

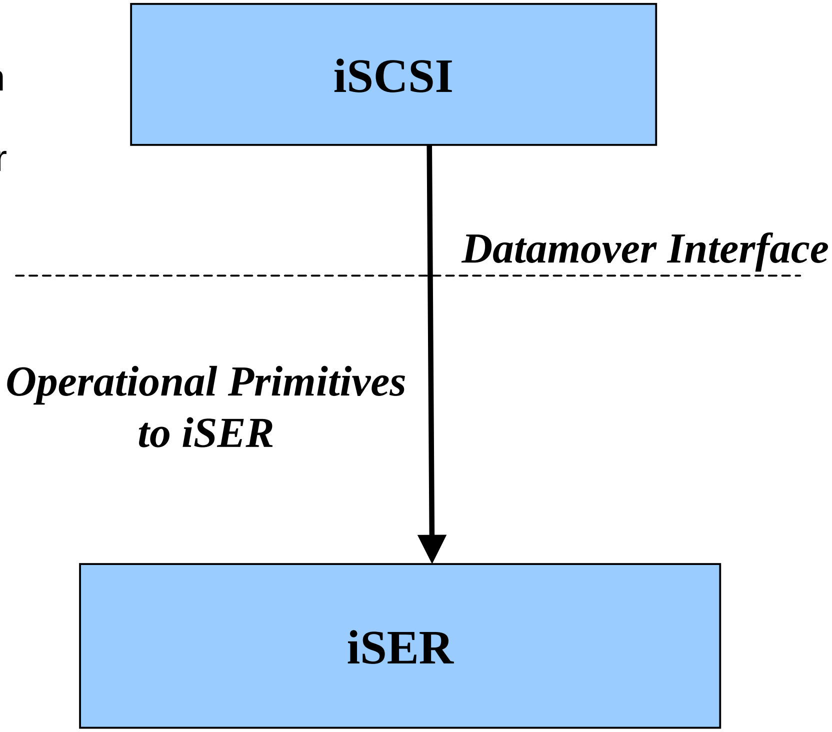


iSER Architectural Features

- iSER extends the data transfer model of the iSCSI protocol
  - Provides iWARP-based data transfer model for iSCSI that enables direct in-order or out-of-order data placement of SCSI data into pre-allocated SCSI buffers while maintaining in-order data delivery
    - Eliminates the memory-to-memory copying overhead incurred in protocol processing, particularly at the receiver, as network speeds grow to 10Gb/s and beyond
    - Requires no iSCSI or iSER specific assists in the iWARP protocol suite or RNIC
  - Simplifies certain protocol aspects of iSCSI such as data integrity management and some error recovery features
- At the same time, iSER maintains compliance with the existing iSCSI protocol
  - Requires no changes to SCSI Architecture Model (SAM/SAM-2/SAM-3) and SCSI Command set standards
  - Utilizes existing iSCSI infrastructure including but not limited to MIB, bootstrapping, negotiation, naming and discovery, and security
  - Utilizes a compatible iSCSI mechanism (login key negotiation) to determine iSER support at the initiator and the target
  - Requires a connection to continue with the semantics as defined in iSCSI if iSER is not supported by either the initiator or the target
    - Therefore, requires no full feature phase interoperability between an end node operating in iSCSI mode, and an end node operating in iSER-assisted mode

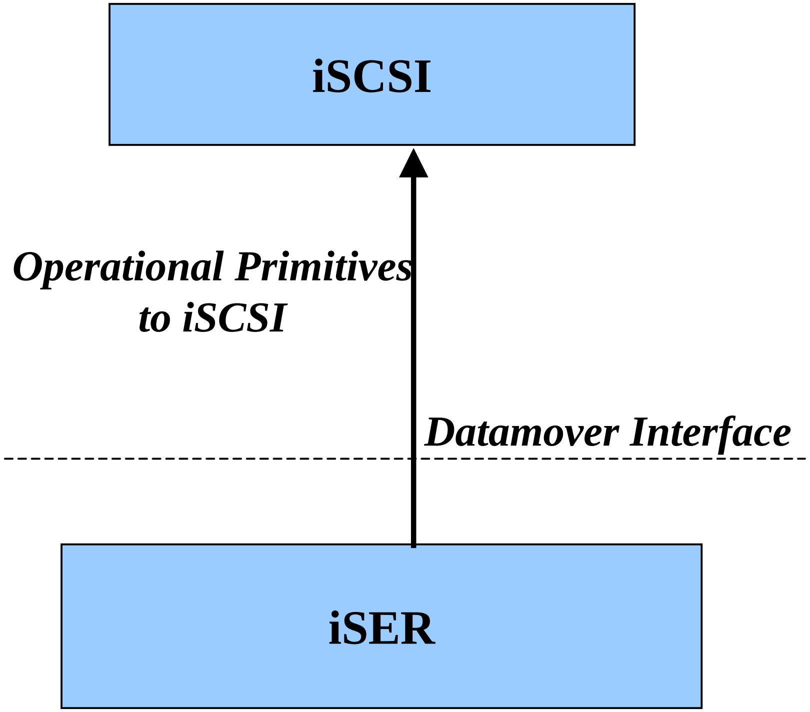
Operational Primitives Provided by the iSER Layer

- These are abstract functional interface procedures that allows the iSCSI layer to request the iSER layer to perform a specific action
- Operational Primitives that can be invoked by the iSCSI layer
  - **Send_Control** requests the outbound transfer of an iSCSI control-type PDU
  - **Put_Data** requests the outbound transfer of data for a SCSI Data-in PDU (target only)
  - **Get_Data** requests the inbound transfer of solicited data requested by an R2T PDU (target only)
  - **Allocate_Connection_Resources** requests the allocation of all iWARP-specific connection resources required for an operational iSCSI/iSER connection
  - **Deallocate_Connection_Resources** requests the deallocation of all iWARP-specific connection resources
  - **Enable_Datamover** requests that a specific iSCSI connection be transitioned to the iSER-assisted mode
  - **Connection_Terminate** requests that a specified iSCSI/iSER connection be terminated and all associated connection and task resources be freed
  - **Notice_Key_Values** requests that the specified Key-Value pairs are to be taken note of by the iSER layer
  - **Deallocate_Task_Resources** requests the deallocation of all iWARP-specific task resources



Operational Primitives Provided by the iSCSI Layer

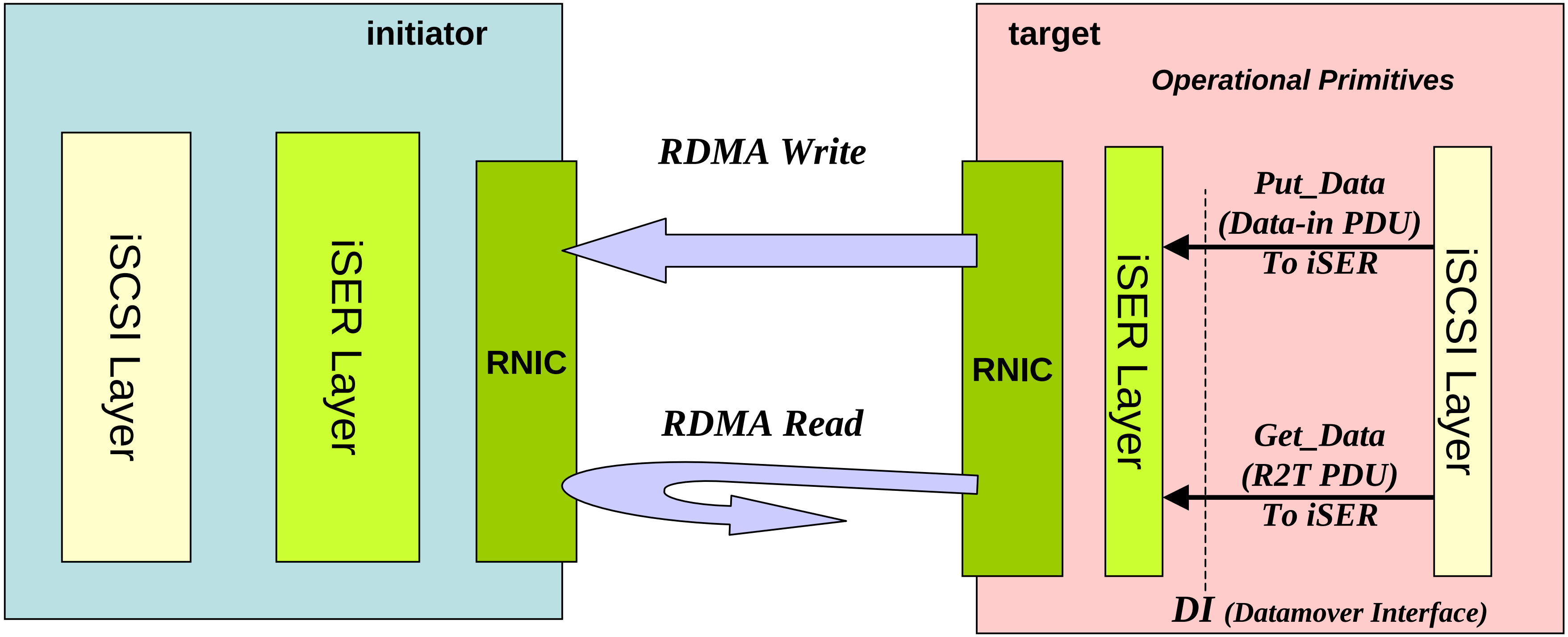
- These are abstract functional interface procedures that allows the iSER layer to notify the iSCSI layer of some event
- Operational Primitives that can be invoked by the iSER layer
  - **Control_Notify** notifies the iSCSI layer of the availability of an inbound iSCSI control-type PDU (see slide 10)
  - **Data_Completion_Notify** notifies the iSCSI layer of the completion of inbound/outbound data transfer that was requested by the iSCSI layer when the request was qualified with Notify_Enable set (target only)
  - **Data_ACK_Notify** notifies the iSCSI layer of the arrival of the data acknowledgement (target only)
  - **Connection_Terminate_Notify** notifies the iSCSI layer of the termination of an iSCSI/iSER connection



iSCSI Data -Type PDUs

- **iSCSI PDUs initiating data transfer into named buffers in the full feature phase are transformed into RDMA Read/Write Messages**
  - Defined as iSCSI data-type PDUs
  - Data transfer is managed by the RNIC hardware/firmware with no involvement from the iSCSI/iSER layers at the initiator or the target during the actual transfer
  - Include the following iSCSI PDUs only
    - **R2T** is transformed into RDMA Read operation by the target
    - **SCSI Data-in** is transformed into RDMA Write operation by the target

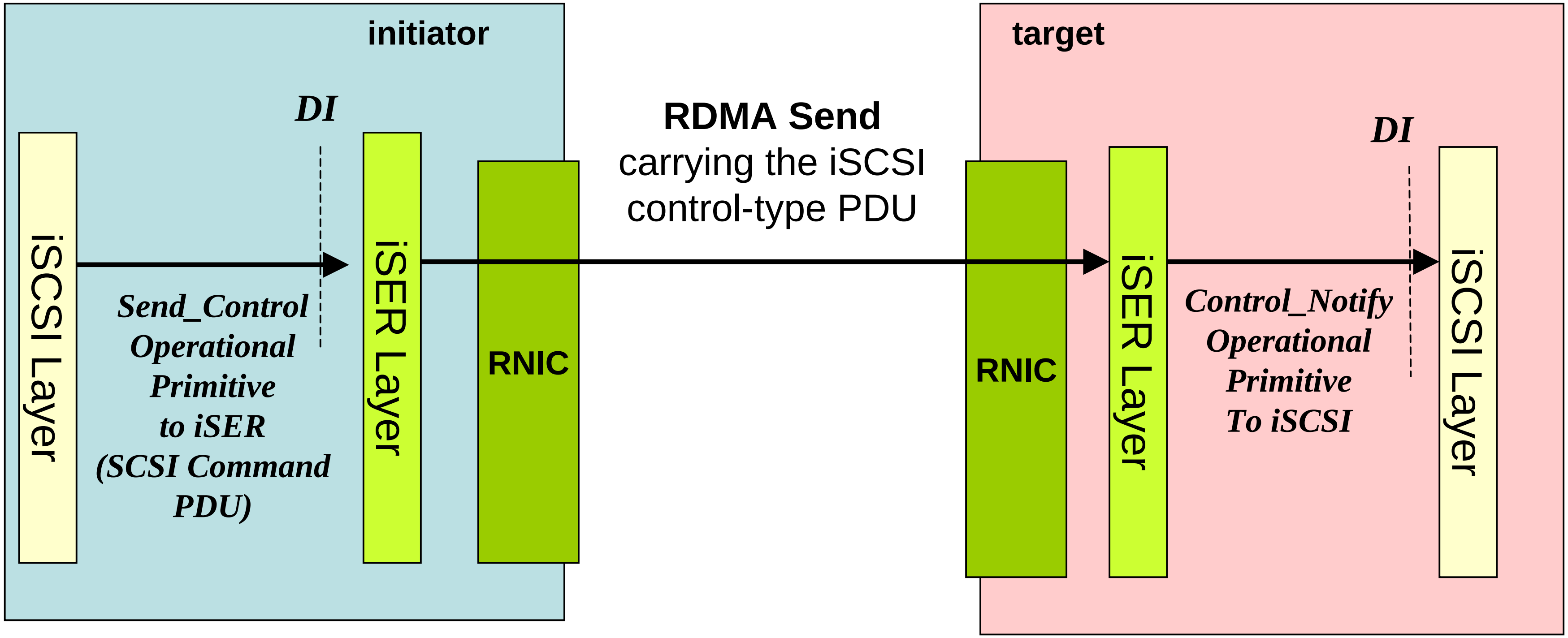
Example of iSCSI Data-Type PDUs



iSCSI Control Type PDUs

- All other iSCSI PDUs in the full feature phase are encapsulated in RDMA Send Type Messages
  - Defined as iSCSI control-type PDUs
  - iSCSI layers at the sending and the receiving nodes are involved in the PDU transfer

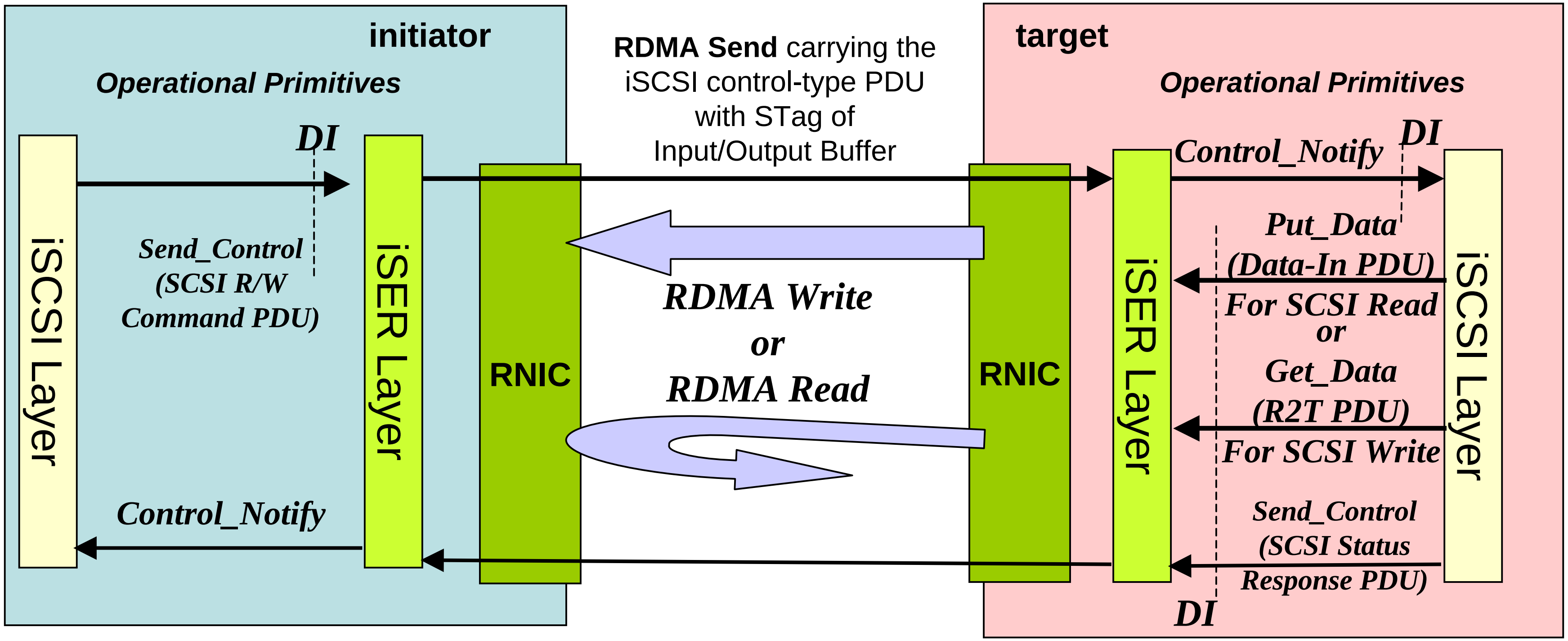
Example of sending iSCSI's SCSI Command PDU


initiator
DI
iSCSI Layer
Send_Control Operational Primitive to iSER (SCSI Command PDU)
iSER Layer
RNIC
RDMA Send
carrying the iSCSI control-type PDU
target
RNIC
iSER Layer
DI
Control_Notify Operational Primitive To iSCSI
iSCSI Layer

iSER Actions for SCSI Read/Write

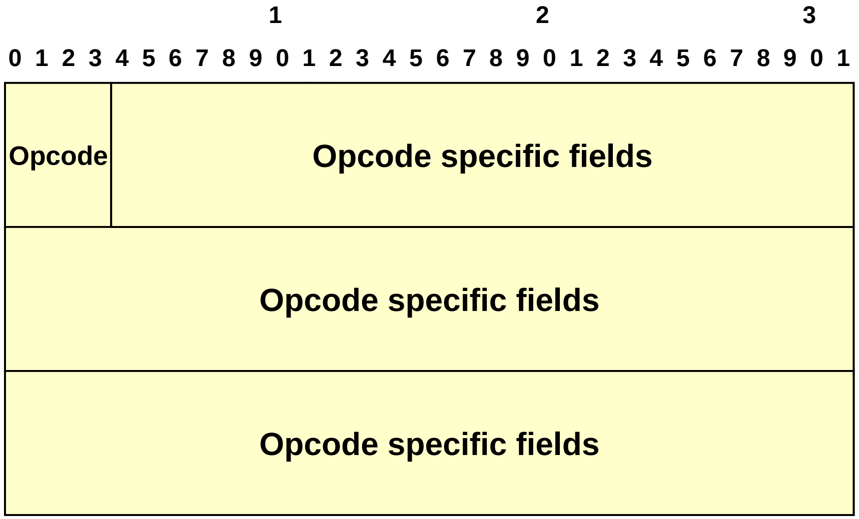
- **iSER layer at the initiator sends (advertises) the buffer identifier (STag(s)) to the target when the SCSI Command for the data-type PDU is issued by the iSCSI layer**
  - For a SCSI Read Command, the STag identifies the tagged buffer into which data from the target will be directly placed by the initiator RNIC using the RDMA Write operation
  - For a SCSI Write Command, the STag identifies the tagged buffer on the initiator from which data is directly fetched by the target RNIC using the RDMA Read operation

Example of SCSI Read or SCSI Write



iSER Header

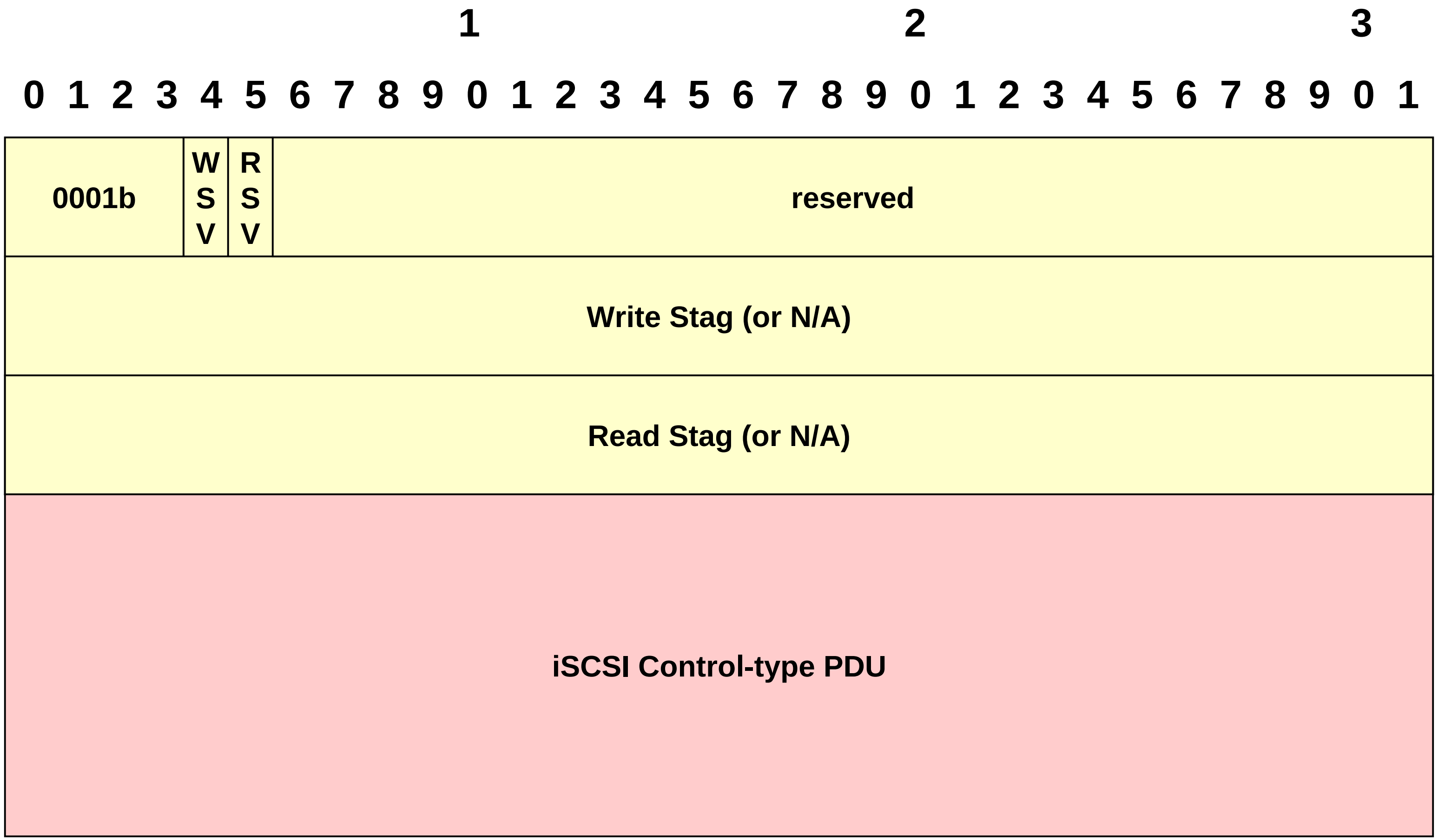
- The iSER header is fixed in size (12 bytes) and is present in all RDMA Send Type Messages for sending iSCSI control-type PDUs and iSER Hello/HelloReply Messages
- The iSER header serves the following purposes:
  - It provides a mechanism for the initiator to advertise the STag(s) for the tagged buffer to the target when a SCSI Command is issued by the iSCSI layer
    - The STags are used later by the target when it transforms the iSCSI data-type PDUs associated the SCSI Command into RDMA Read or RDMA Write operations
  - It allows the iSER layers at the initiator and the target to exchange operational parameters for the connection during connection setup
- The iSER header, when present, immediately follows the iWARP header (not shown)
- A 4-bit opcode field determines the layout of the iSER header



•Opcode values:

- **•0001b = iSCSI control-type PDU**
- **•0010b = iSER Hello Message**
- **•0011b = iSER HelloReply Message**

iSER Header for iSCSI control-type PDU



WSV = Write STag valid
RSV = Read STag valid

Example of an iSER/iWARP Message Carrying a SCSI Command PDU of Type SCSI Read

<table>
<tr><td colspan="4">MPA Header</td><td>DDP Control</td><td>RDMAP Control</td></tr>
<tr><td colspan="6">RsvdULP (value = 0)</td></tr>
<tr><td colspan="6">(Send) Queue Number (value=0)</td></tr>
<tr><td colspan="6">(Send) Message Sequence Number</td></tr>
<tr><td colspan="6">(Send) Message Offset</td></tr>
<tr><td>0001’b</td><td>0</td><td>1</td><td colspan="3">Reserved (value = 0)</td></tr>
<tr><td colspan="6">Write STag (value = 0)</td></tr>
<tr><td colspan="6">Read STag</td></tr>
<tr><td colspan="6">iSCSI’s SCSI Command PDU (Read)</td></tr>
<tr><td colspan="6">MPA CRC</td></tr>
</table>

Note: Markers, if any, are not shown

iSER Header for ISER Hello Message

```
                     1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
```

0010b	reserved	Maximum Version	Minimum Version	ISER-IRD
reserved				
reserved				

iSER Header for iSER HelloReply Message

<table>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>3</td><td></td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td></tr>
<tr><td colspan="4">0011b</td><td colspan="3">reserved</td><td colspan="1">REJ</td><td colspan="4">Maximum Version</td><td colspan="4">Current Version</td><td colspan="16">iSER-ORD</td></tr>
<tr><td colspan="32">reserved</td></tr>
<tr><td colspan="32">reserved</td></tr>
</table>

- **REJ – Connection is rejected, if set to 1**

Example of an iSER Hello Message in an iWARP Send Message

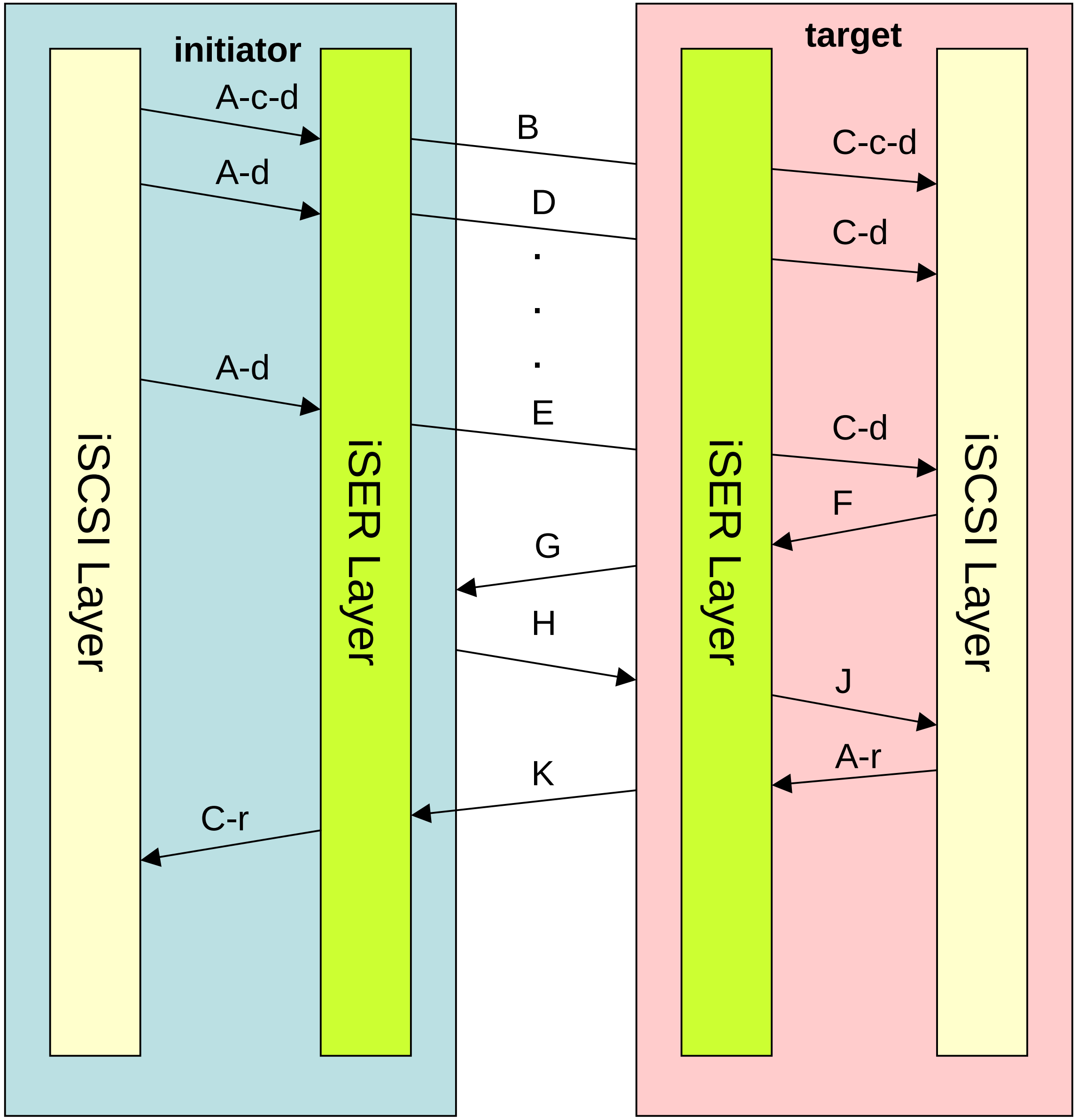
<table>
<tr><td colspan="4">MPA Header</td><td>DDP Control</td><td>RDMAP Control</td></tr>
<tr><td colspan="6">RsvdULP (value = 0)</td></tr>
<tr><td colspan="6">(Send) Queue Number (value=0)</td></tr>
<tr><td colspan="6">(Send) Message Sequence Number</td></tr>
<tr><td colspan="6">(Send) Message Offset</td></tr>
<tr><td>0010’b</td><td>Reserved (values = 0)</td><td>0001’b</td><td>0001’b</td><td colspan="2">iSER-IRD</td></tr>
<tr><td colspan="6">Reserved (value = 0)</td></tr>
<tr><td colspan="6">Reserved (value = 0)</td></tr>
<tr><td colspan="6">MPA CRC</td></tr>
</table>

Note: Markers, if any, are not shown

SCSI Write Operation

- The iSCSI layer at the initiator invokes the Send_Control Primitive to request the iSER layer to send the SCSI Command PDU
  - The iSER layer requests the RDMAP layer to transmit a Send Message containing the SCSI Command PDU and immediate data (if any)
  - If there is solicited data, the iSER layer at the initiator advertises the Write STag in the iSER header in the Send Message
- Upon receiving the Send Message containing the SCSI Command PDU, the iSER layer at the target notifies the iSCSI layer using the Control_Notify Primitive
- If there is non-immediate unsolicited data, the iSCSI layer at the initiator invokes the Send_Control Primitive to request the iSER layer to send the SCSI Data-out PDU
- Upon receiving the Send Message containing the SCSI Data-out PDU, the iSER layer at the target will notify the iSCSI layer using the Control_Notify Primitive
- If there is solicited data, the iSCSI layer at the target invokes the Get_Data Primitive when it has an I/O buffer available to request the iSER layer to handle the R2T PDU
  - The iSER layer at the target transforms each R2T into an RDMA Read Operation
- If requested by the iSCSI layer during the invocation of Get_Data, the iSER layer will notify the iSCSI layer at the target using the Data_Completion_Notify Primitive upon the completion of the RDMA operation
- Upon completing the data transfer, the iSCSI layer at the target invokes the Send_Control Primitive to request the iSER layer to send the SCSI Response PDU. The iSER layer passes the STag in the RDMA Send with Invalidate Message.
- Upon receiving the Send with Invalidate Message containing the SCSI Response PDU, the RNIC at the initiator invalidates the STag and notifies the iSCSI layer of the iSCSI PDU using the Control_Notify Primitive

Example of SCSI Write Data Transfer



A. Send_Control to send SCSI PDU
    c - command
    d -data
    r - response
B. iWARP Send Message
    containing SCSI Command
    PDU with immediate data
C. Control_Notify to report SCSI
    PDU received
    c – command
    d – data
    r – response
D. iWARP Send Message
    containing SCSI Data-out
    PDU with unsolicited data
E. iWARP Send Message
    containing SCSI Data-out
    PDU with last unsolicited data
    segment
F. Get_Data for R2T
G. iWARP RDMA Read Request (*)
H. iWARP RDMA Read Response
    with solicited data (*)
J. Data_Completion_Notify
K. iWARP Send with Invalidate
    Message containing SCSI
    Response PDU

* RDMA Read data transfer is handled by the RNIC

Example of an RDMA Read Response Message Containing SCSI Write Data

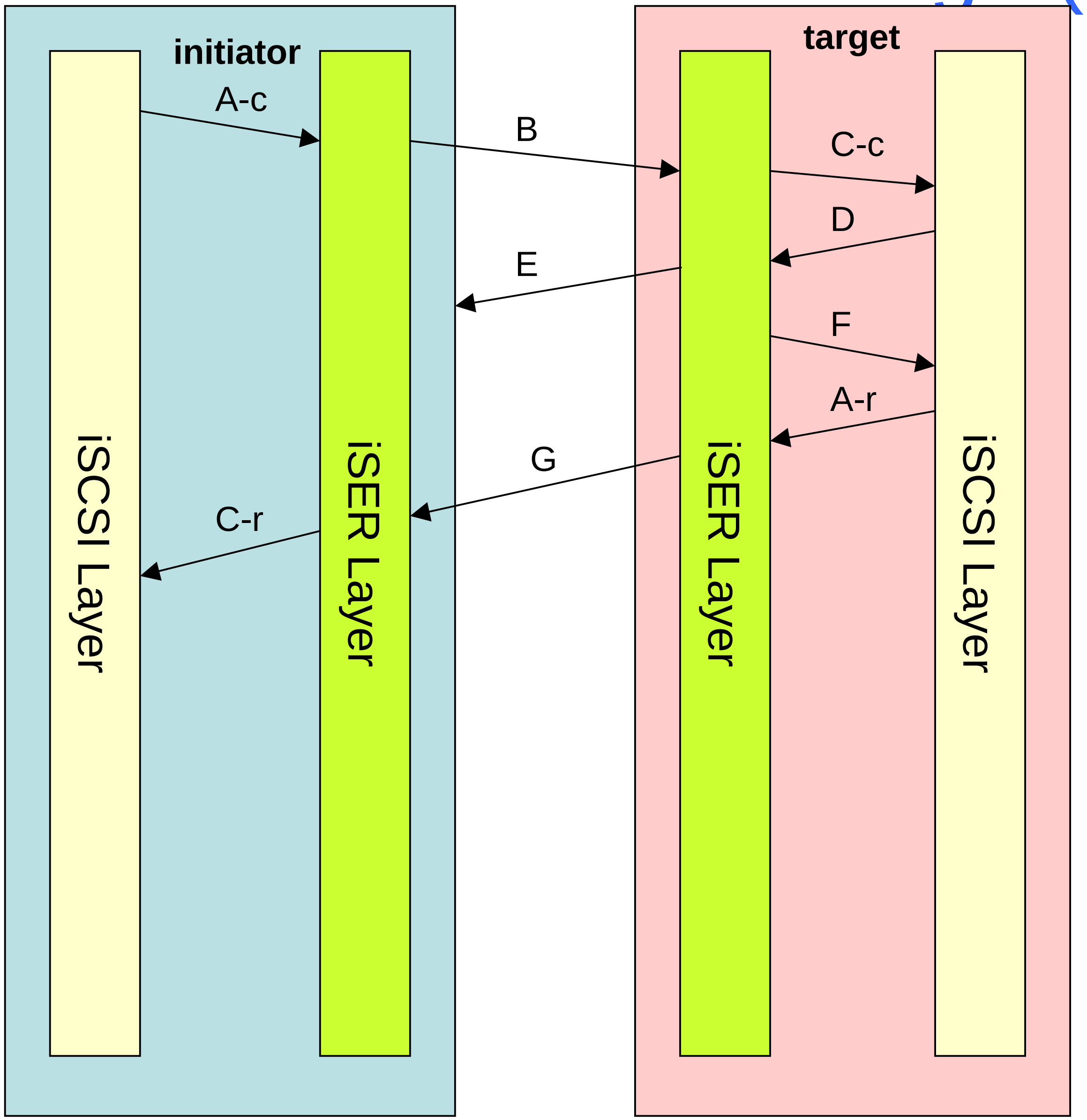
<table>
<tr><td colspan="2">MPA Header</td><td>DDP Control</td><td>RDMAP Control</td></tr>
<tr><td colspan="4">Data Sink STag</td></tr>
<tr><td colspan="4">Data Sink Tagged Offset</td></tr>
<tr><td colspan="4">SCSI Write Data</td></tr>
<tr><td colspan="4">MPA CRC</td></tr>
</table>

Note: Markers, if any, are not shown

SCSI Read Operation

- The iSCSI layer at the initiator invokes the Send_Control Primitive to request the iSER layer to send the SCSI Read Command PDU.
- The iSER layer requests the RDMAP layer to transmit a Send Message containing the SCSI Command PDU
  - The iSER layer advertises the Read STag in the iSER header in the Send Message
- Upon receiving the Send Message containing the SCSI Command PDU, the iSER layer at the target notifies the iSCSI layer using the Control_Notify Primitive
- When the requested data is available at the I/O buffer, the iSCSI layer at the target invokes the Put_Data Primitive to request the iSER layer to handle the SCSI Data-in PDU
  - The iSER layer at the target transforms the SCSI Data-in PDU into an RDMA Write operation
- If requested by the iSCSI layer during the invocation of Put_Data, the iSER layer will notify the iSCSI layer at the target using the Data_Completion_Notify Primitive upon the completion of the RDMA Write operation
- If the A-bit is set in the SCSI Data-in PDU, the iSER layer at the target will notify the iSCSI layer when the data transfer is complete at the initiator
  - If ErrorRecoveryLevel is 2 or unknown, the iSER layer at the target will issue a zero-length RDMA Read operation and notifies the local iSCSI layer upon the completion of the RDMA Read operation
- Upon completing the data transfer, the iSCSI layer at the target invokes the Send_Control Primitive to request the iSER layer to send the SCSI Response
  - SCSI status is always returned in a separate SCSI Response PDU (“Phase collapse” in SCSI Read Command is not allowed)
  - The iSER layer passes the STag in the RDMA Send with Invalidate Message
- Upon receiving the Send with Invalidate Message containing the SCSI Response PDU, the RNIC at the initiator invalidates the STag and the iSER layer notifies the local iSCSI layer using the Control_Notify Primitive

Example of SCSI Read Data Transfer with no Acknowledge (A bit = 0)



A. Send_Control to send SCSI PDU
c – command
r - response

B. iWARP Send Message containing SCSI Command PDU

C. Control_Notify to report SCSI PDU received
c – command
r - response

D. Put_Data for SCSI Data-in PDU

E. iWARP RDMA Write Message (*)

F. Data_Completion_Notify

G. iWARP Send with Invalidate Message containing SCSI Response PDU

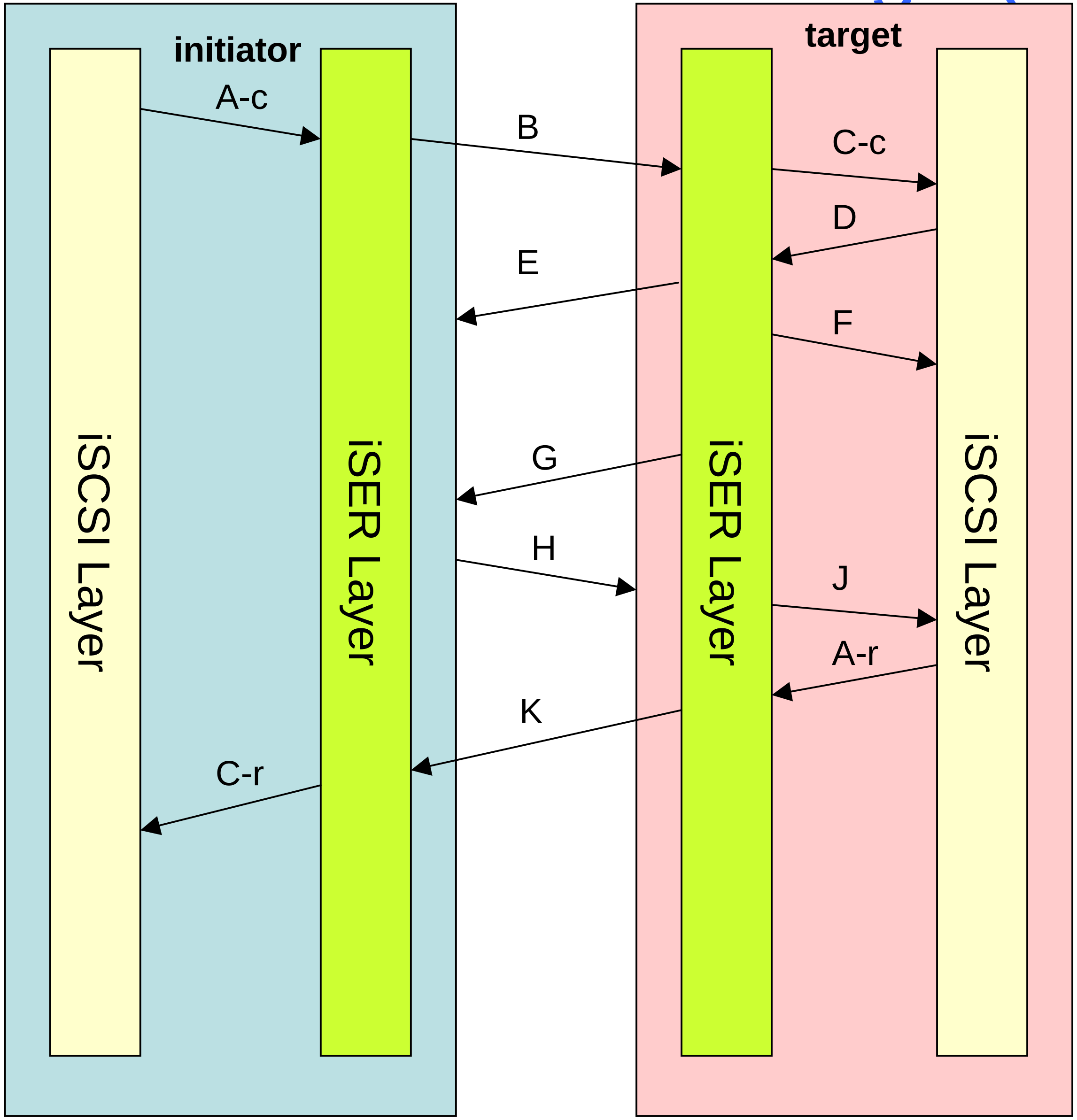
* RDMA Write data transfer is handled by the RNIC

Example of an RDMA Write Message Containing SCSI Read Data

<table>
<tr><td><b>MPA Header</b></td><td><b>DDP Control</b></td><td><b>RDMAP Control</b></td></tr>
<tr><td colspan="3"><b>Data Sink STag</b></td></tr>
<tr><td colspan="3"><b>Data Sink Tagged Offset</b></td></tr>
<tr><td colspan="3"><b>SCSI Read Data</b></td></tr>
<tr><td colspan="3"><b>MPA CRC</b></td></tr>
</table>

Note: Markers, if any, are not shown

Example of SCSI Read Data Transfer with Acknowledge (A bit = 1)



A. Send_Control to send SCSI PDU
    c – command
    r - response

B. iWARP Send Message containing SCSI Command PDU

C. Control_Notify to report SCSI PDU received
    c – command
    r - response

D. Put_Data to process SCSI Data-in PDU

E. iWARP RDMA Write Message (*)

F. Optional Data_Completion_Notify

G. iWARP RDMA Read Request with zero length

H. iWARP RDMA Read Response with zero length

J. Data_ACK_Notify to report data received at initiator

K. iWARP Send with Invalidate Message containing SCSI Response PDU

* RDMA Write data transfer is handled by the RNIC

Flow Control

- For RDMA Send Type Messages
  - The iSER protocol does not provide additional flow control beyond that provided by the iSCSI layer on control-type PDUs
  - An implementation should be able to take advantage of iWARP Verbs mechanisms such as the Shared Receive Queue mechanism to effectively address the Send Message flow control question
- For RDMA Read Resources
  - In the iSER Hello Message, the iSER layer at the initiator declares the maximum number of RDMA Read Requests that the initiator can receive on the particular RDMAP Stream (iSER-IRD) to the target
    - This allows the iSER layer at the target to adjust its resources if it can issue more RDMA Read Requests than the initiator can handle
  - In the iSER HelloReply Message, the iSER layer at the target declares the maximum number of RDMA Read Requests that the target can issue on a particular RDMAP Stream (iSER-ORD) to the initiator
    - This allows the iSER layer at the initiator to adjust its resources if it can handle more RDMA Read Requests than the target can issue
  - The iSER layer at the target will flow control the RDMA Read Request Messages to not exceed iSER-ORD

General Considerations for an iSCSI/iSER Connection Setup

- During connection setup, the iSCSI layer at the initiator is responsible for establishing a TCP connection
  - Use the TCP port as discovered through the iSCSI discovery mechanisms
- iSCSI Login negotiation follows the same rules as in the iSCSI specification with the following changes:
  - The iSCSI layers at the initiator and the target negotiate the new RDMAExtensions key on the leading connection in order to enable iSER-assisted mode
  - Header Digest and Data Digests are negotiated to “None”
    - Data integrity is already provided by the MPA CRC
    - Managing the digests in the RNIC would mean that the RNIC have to be ULP-aware
  - The iSCSI layer negotiates the new TargetRecvDataSegmentLength key and the InitiatorRecvDataSegmentLength key for each connection
    - Whenever the initiator (or target) sends an iSCSI control-type PDU to the target (or initiator), non-final PDUs have a data segment size of exactly TargetRecvDataSegmentLength (or InitiatorRecvDataSegmentLength)
  - OFMarker and IFMarker are negotiated to “No”
    - Markers are provided by the MPA layer
    - Managing the iSCSI markers would mean that the RNIC have to be ULP-aware

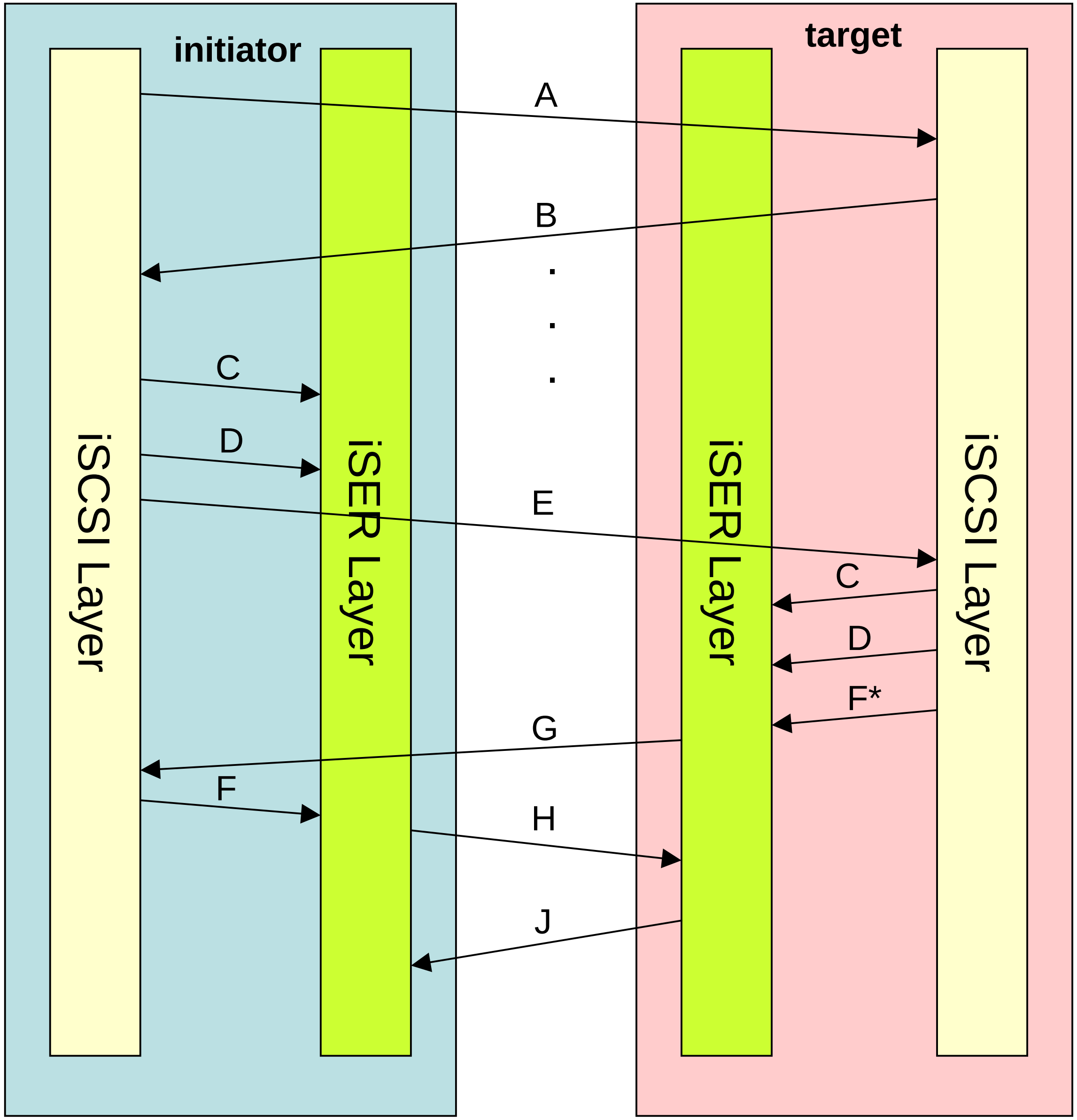
Connection Setup for iSER-assisted Mode at the Initiator

- Negotiated key values may be passed by the iSCSI layer to the iSER layer by invoking the Notice_Key_Values Operational Primitive
- Before sending the final Login Request, the iSCSI layer invokes the Allocate_Connection_Resources Operational Primitive to request the iSER layer to allocate the iWARP resources for the connection
- After the target returns the final Login Response, the iSCSI layer at the initiator invokes the Enable_Datamover Operational Primitive to request the iSER layer to transition into iSER-assisted mode
- The first message sent by the iSER layer at the initiator to the target is the iSER Hello Message

Connection Setup for iSER-assisted Mode at the Target

- Negotiated key values may be passed by the iSCSI layer to the iSER layer by invoking the Notice_Key_Values Operational Primitive
- Before sending the final Login Response, the iSCSI layer invokes the Allocate_Connection_Resources Operational Primitive to request the iSER layer to allocate the iWARP resources for the connection
- The iSCSI layer invokes the Enable_Datamover Operational Primitive to enable the iSER mode qualified with the final Login Response PDU
- The iSER layer sends the final Login Response PDU in byte stream mode and then transitions into iSER-assisted mode
- After receiving the iSER Hello Message from the initiator, the iSER layer at the target responds by sending the iSER HelloReply Message

Example of Successful iSER Connection Setup



A. SCSI Login Request PDU with RDMAExtensions=Yes

B. SCSI Login Response PDU with RDMAExtensions=Yes

C. Optional Notice_Key_Values to pass values of negotiated keys

D. Allocate_Connection_Resources to set up iWARP resources

E. SCSI Login Request PDU with T=1 and NSG=FullFeaturePhase

F. Enable_Datamover to go into iSER mode (* = send last iSCSI PDU in byte stream mode)

G. SCSI Login Response PDU in byte stream mode with T=1 and NSG=FullFeaturePhase

H. iWARP Send Message containing iSER Hello

J. iWARP Send Message containing iSER HelloReply

Unsuccessful Connection Setup for iSER-assisted Mode

- During the login phase, if both the initiator and the target fail to negotiate the RDMAExtensions key to “Yes”, then the connection continues with the semantics as defined in iSCSI

s defined in iSCSI

r reason it is not possible to enter iSER-assisted mode, the connection will

will be terminated

locating the iWARP resources, the iSCSI layer will terminate the connection

ate the connection

e iWARP resources, the iSCSI layer will send a Login Response with the cessful status and terminate the connection

ate the connection

in the iSER Hello Message is unacceptable to the target, the iSER layer will set the Reject flag in the iSER HelloReply Message and terminate the RDMAP

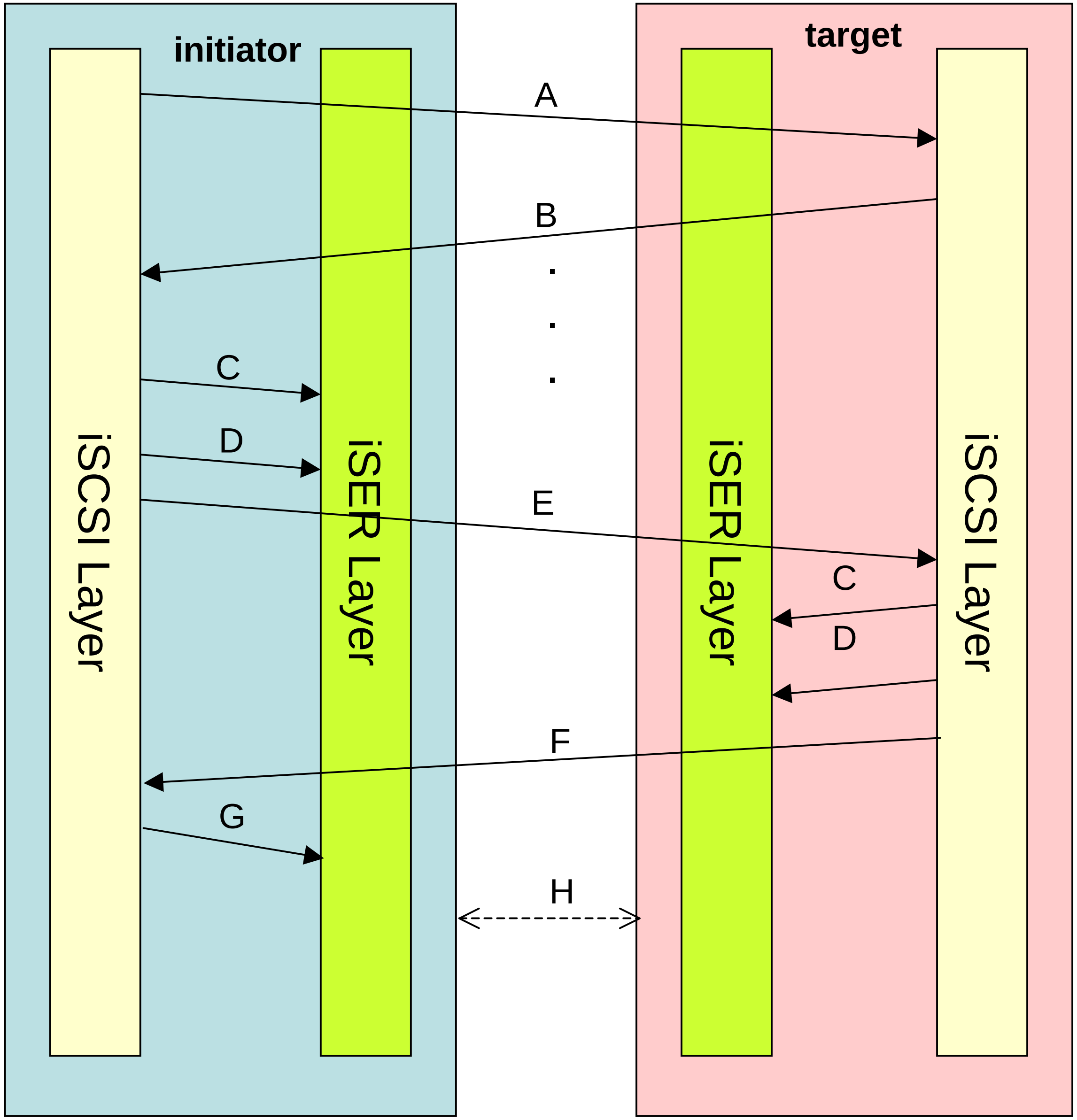
e the RDMAP stream

n_Terminate_Notify Operational Primitive after the RDMAP stream is terminated and all

resources for the connection have been released

have been released

Example of Unsuccessful iSER Connection Setup



A. SCSI Login Request PDU with RDMAExtensions=Yes

B. SCSI Login Response PDU with RDMAExtensions=Yes

C. Optional Notice_Key_Values to pass values of negotiated keys

D. Allocate_Connection_Resources to set up iWARP resources

E. SCSI Login Request PDU with T=1 and NSG=FullFeaturePhase

F. SCSI Login Response PDU with unsuccessful status

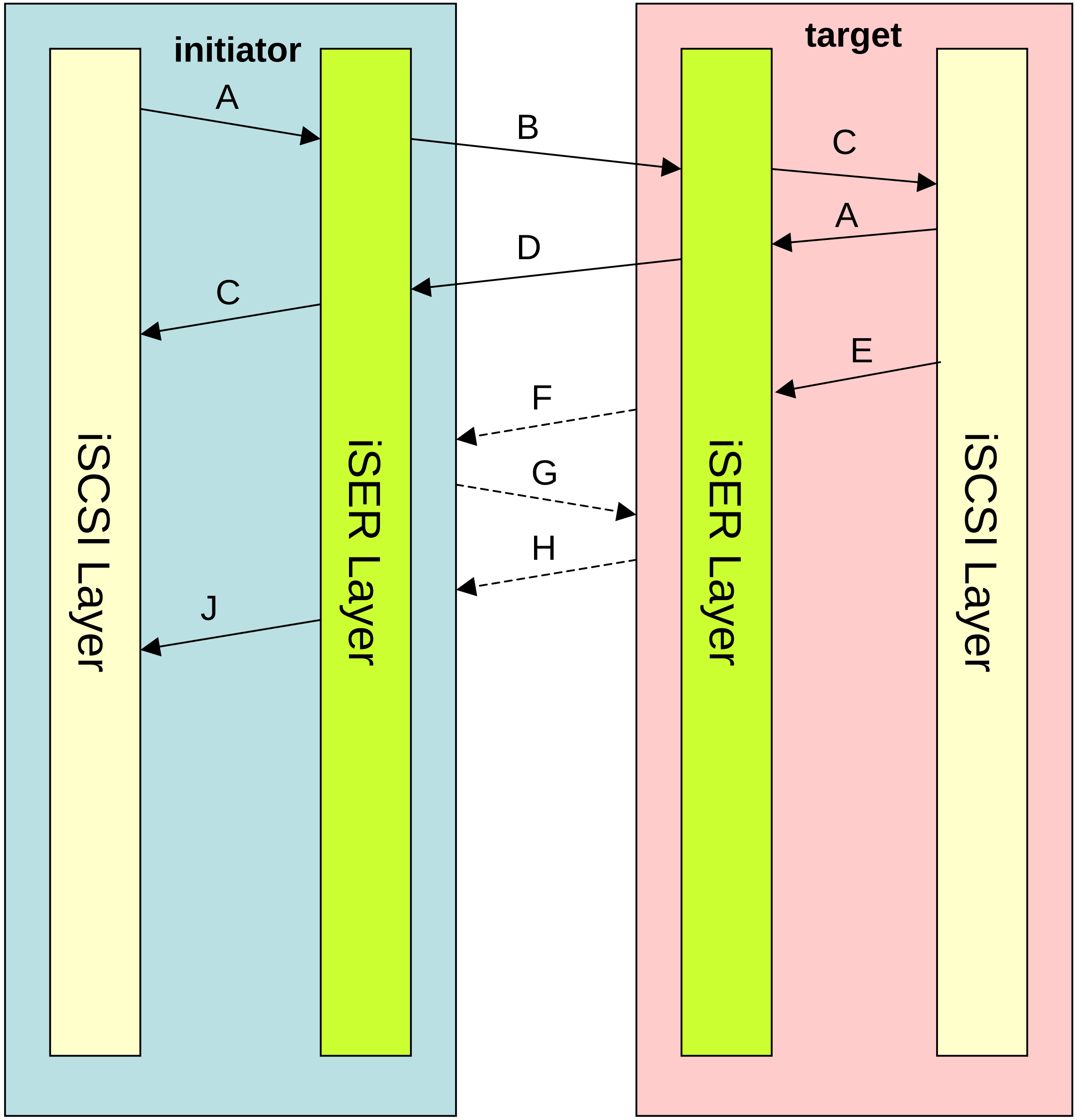
G. Deallocate_Connection_Resources to release iWARP resources

H. TCP FIN exchanges to close the connection

Normal Connection Termination for iSER-assisted Mode

- The iSCSI layer at the initiator invokes the Send_Control Operational Primitive to request the iSER layer to send the Logout Request PDU
- The iSER layer at the target notifies the iSCSI layer of the Logout Request PDU by invoking the Control_Notify Operational Primitive
- The iSCSI layer at the target invokes the Send_Control Operational Primitive to request the iSER layer to send the Logout Response PDU
- The iSER layer at the initiator notifies the iSCSI layer of the Logout Response PDU by invoking the Control_Notify Operational Primitive
- After completing the iSCSI logout process, the iSCSI layer at the target invokes the Connection_Terminate Operational Primitive to request the iSER layer to terminate the RDMAP stream and release all resources for the connection
- After the TCP connection has been closed, the iSER layer at the initiator releases all resources for the connection and notifies the iSCSI layer by invoking the Connection_Terminate_Notify Operational Primitive

Example of Normal iSER Connection Termination



A. Send_Control to send SCSI PDU

B. iWARP Send Message
containing SCSI Logout
Request PDU

C. Control_Notify to report SCSI
PDU received

D. iWARP Send Message
containing SCSI Logout
Response PDU

E. Connection_Terminate to close
the connection and release
iWARP resources

F. TCP FIN

G. TCP FIN + Ack

H. TCP Ack

J. Connection_Terminate_Notify to
report that the connection is
closed

Error Recovery

- All three ErrorRecoveryLevels as defined in iSCSI may be deployed in the iSER-assisted mode
    - The following considerations were made in order to support ErrorRecoveryLevels 1 and 2
- For ErrorRecoveryLevel 1
    - The iSCSI layer at the initiator should disable timeout-driven proactive SNACKs and timeout-driven PDU retransmissions since digest and sequence errors will not occur in the iSER-assisted mode
    - The PDU recovery realized via this ErrorRecoveryLevel will never be necessary since digests are not used and hence may be considered always supported
- For ErrorRecoveryLevel 2
    - When the iSCSI layer at the target accepts a reassignment request for a SCSI Read Command, it will invoke the Put_Data Operational Primitive to request the iSER layer to process the SCSI Data-in PDU if not all data is acknowledged
        - The iSCSI layer at the initiator will set ExpDataSN = 0 on Task Allegiance Reassignment to allow the target to request all unacknowledged data
    - When the iSCSI layer at the target accepts a reassignment request for a SCSI Write Command, it will invoke the Get_Data Operational Primitive to request the iSER layer to process the R2T PDU for any non-immediate unsolicited data and any solicited data that have not been received
        - Data previously designated as unsolicited will also be transferred using RDMA Read operations
- Note that iSCSI data acknowledgement support for ErrorRecoveryLevels 1 and 2 are described in slide 24

Summary

- The iSCSI Extensions for RDMA (iSER) is based on the Datamover Architecture. The iSER protocol allows the data movement and placement aspects of iSCSI to be offloaded to the iWARP protocol suite
  - Provides the option of using generic RNICs for iSCSI instead of dedicated iSCSI HBAs
    - Requires no iSCSI or iSER specific assists in the iWARP protocol suite or RNIC
  - Enables direct data placement of in-order or out-of-order SCSI data into pre-allocated SCSI buffers
    - Eliminates the data copy in the receive path to move the data to the final buffer
    - Eliminates unnecessary memory bandwidth consumption
    - Decreases reassembly buffer size requirements
    - Reduces CPU utilization
- iSER requires no changes to SCSI Architecture Model (SAM/SAM-2/SAM-3) and SCSI Command set standards
- iSER fully utilizes existing iSCSI infrastructure including but not limited to MIB, bootstrapping, negotiation, naming and discovery, and security
- iSER seeks to minimize impacts to existing iSCSI implementations in supporting the extensions
- For additional information, go to the RDMA Consortium website at www.rdmaconsortium.org