bluegenep	CMK_MSG_HEADER_BASIC_ CmiUInt4 size; CmiUInt2 hdl, rank; unsigned char cksum, magic; ./bluegenep/conv-common.h:#define CMK_MSG_HEADER_BASIC { CMK_MSG_HEADER_BASIC_ CmiUInt2 pad[3]; } ./bluegenep/conv-common.h:#define CMK_MSG_HEADER_BASIC_ CmiUInt4 size; CmiUInt2 hdl, rank; ./bluegenep/conv-common.h:#define CMK_MSG_HEADER_BASIC { CMK_MSG_HEADER_BASIC_ } ./bluegenep/conv-common.h:#define CMK_MSG_HEADER_EXT_ CMK_MSG_HEADER_BASIC_ CmiUInt2 stratid, xhdl, info, redID; CmiInt4 root; CmiUInt2 pad1; ./bluegenep/conv-common.h:#define CMK_MSG_HEADER_EXT { CMK_MSG_HEADER_EXT_ CmiUInt2 pad1;} ./bluegenep/conv-common.h:#define CMK_MSG_HEADER_EXT { CMK_MSG_HEADER_EXT_ CmiUInt4 pad1;} ./bluegenep/conv-common.h:#define CMK_MSG_HEADER_BIGSIM_ { CMK_MSG_HEADER_EXT_ CMK_BIGSIM_FIELDS }
uth	CmiUInt2 hdl,xhdl,info,stratid,root,redID,pad2,pad3;
verbs	CmiUInt2 d0,d1,d2,d3,d4,d5,hdl,stratid,root,xhdl,info,redID, rank;
./verbs/conv-mach-syncft.h:	CmiUInt2 d0,d1,d2,d3,d4,d5,hdl,pn,d8,stratid,root,xhdl,info,dd,redID,pad2;
shmem	char gap[56]; CmiUInt2 hdl,xhdl,info,stratid,root,redID,padding2,padding3;
./netlrts/conv-mach-syncft.h:	CMK_MSG_HEADER_EXT_ CmiUInt2 d0,d1,d2,d3,d4,d5,hdl,pn,d8,stratid,root,xhdl,info,dd,redID,pad2;
netlrts	CMK_MSG_HEADER_EXT_ CmiUInt2 d0,d1,d2,d3,d4,d5,hdl,stratid,root,xhdl,info,redID, rank;
./pamilrts/conv-mach-async.h	#define CMK_MSG_HEADER_EXT_ CmiUInt2 rank, hdl,xhdl,info, stratid; unsigned char cksum, magic; int root, size, dstnode; CmiUInt2 redID, padding; char work[6*sizeof(uintptr_t)]; CmiUInt4 compressStart; CmiUInt2 compress_flag,xxhdl; CmiUInt8 persistRecvHandler;
	./pamilrts/conv-mach-async.h:#define CMK_MSG_HEADER_EXT_ CmiUInt2 rank, hdl,xhdl,info, stratid; unsigned char cksum, magic; int root, size, dstnode; CmiUInt2 redID, padding; char work[6*sizeof(uintptr_t)];
pamilrts	CMK_MSG_HEADER_EXT_ CmiUInt2 rank, hdl,xhdl,info, stratid; unsigned char cksum, magic; int root, size; CmiUInt2 redID, padding;
./template/conv-common.h	:#define CMK_MSG_HEADER_UNIQUE CmiUInt4 size; CmiUInt2 hdl,xhdl,info;
./net/conv-mach-syncft.h:	e CMK_MSG_HEADER_BASIC_ CmiUInt2 d0, d1, d2, d3, d4, d5; CmiUInt4 size; CmiUInt2 hdl; ./net/conv-common.h:#define CMK_MSG_HEADER_BASIC { CMK_MSG_HEADER_BASIC_ CmiUInt2 pad1, pad2, pad3; } ./net/conv-common.h:#define CMK_MSG_HEADER_EXT_ CMK_MSG_HEADER_BASIC_ CmiUInt2 stratid, xhdl, info; CmiInt4 root; CmiUInt2 redID;
net	CMK_MSG_HEADER_BASIC_ CmiUInt2 d0, d1, d2, d3, d4, d5; CmiUInt4 size; CmiUInt2 hdl;
	./net/conv-common.h:#define CMK_MSG_HEADER_BASIC { CMK_MSG_HEADER_BASIC_ CmiUInt2 pad1, pad2, pad3; }
	./net/conv-common.h:#define CMK_MSG_HEADER_EXT_ CMK_MSG_HEADER_BASIC_ CmiUInt2 stratid, xhdl, info; CmiInt4 root; CmiUInt2 redID;
	./net/conv-common.h:#define CMK_MSG_HEADER_EXT { CMK_MSG_HEADER_EXT_ CmiUInt2 pad1; }
gni	CMK_MSG_HEADER_EXT_ CmiUInt4 size; CmiUInt2 seq; unsigned char cksum, magic; CmiUInt2 rank,hdl,xhdl,info,stratid,redID; CmiInt4 root; ./gni/conv-common.h:#define CMK_MSG_HEADER_EXT_ CmiUInt4 size; CmiUInt4 seq; CmiUInt2 rank,hdl,xhdl,info,stratid,redID; CmiInt4 root;
./gni/conv-mach-syncft.h	#define CMK_MSG_HEADER_EXT_ CmiUInt4 size; CmiUInt4 seq; CmiUInt2 rank,hdl,xhdl,info,stratid,redID,pn,d9; CmiInt4 root;
lapi	CMK_MSG_HEADER_FIELDS CmiUInt2 rank,hdl,xhdl,info,stratid,redID; int root, size, srcpe, seqno;
elan	CMK_MSG_HEADER_EXT_ CmiUInt4 size; CmiUInt2 rank,root,hdl,xhdl,info,stratid,redID,pad2; CmiUInt4 pad4;
./pami/conv-mach-async.h:	:#define CMK_MSG_HEADER_EXT_ CmiUInt2 rank, hdl,xhdl,info, stratid; unsigned char cksum, magic; int root, size, dstnode; CmiUInt2 redID, padding; char work[6*sizeof(uintptr_t)];
pami	define CMK_MSG_HEADER_EXT_ CmiUInt2 rank, hdl,xhdl,info, stratid; unsigned char cksum, magic; int root, size; CmiUInt2 redID, padding;

mpi	CMK_MSG_HEADER_EXT_ CmiUInt2 rank, hdl,xhdl,info, stratid, redID; CmiInt4 root; unsigned char cksum, magic; ./mpi/conv-common.h:#define CMK_MSG_HEADER_EXT_ CmiUInt2 rank, hdl,xhdl,info, stratid, redID; CmiInt4 root;
./mpi/conv-mach-syncft.h	:#define CMK_MSG_HEADER_EXT_ CmiUInt2 rank, root, hdl,xhdl,info, stratid, pn,d7; unsigned char cksum, magic; CmiUInt2 redID;
bluegenel	CmiUInt2 rank, hdl,xhdl,info, stratid; unsigned char cksum, magic; int root, size; CmiUInt2 redID, padding;
portals-crayxt3	CMK_MSG_HEADER_EXT_ CmiUInt2 rank, root, hdl,xhdl,info, stratid; unsigned char cksum, magic; CmiUInt2 redID;
	1. root is int wheras info and redID are CmiUInt 2
	2. Will need a bit to tell which field of union is in use
	3. Info usage is mostly confined to conv-ldb (CmiGetInfo, CmiSetInfo)
	4. redID used in conv-core/concore.c for reduction, also used directly by conv-ccs (CmiGetRedID, CmiSetRedID)
	5 root used for broadcast (CMI_BROADCAST_ROOT, CMI_SET_BROADCAST_ROOT, CmiGetRoot, CmiSetRoot)
	6. Will need 2 bits to tell which of this fields is valid, code will need to be modified to check accordingly
	7. Code modification to access everything as a union member (and some code paths which first check the bit field

size (int)	d0 d1 d2 d3 etc		
hdl			
rank	CmiUInt2		
stratid	CmiUInt2	Commlib strategy id	
xhdl	CmiUInt2	Extended handler	
redID	CmiUInt2	Reduction ID	
root	int	Broadcast root (also used to check if the message is broadcast or not)	
info	CmiUInt2	used with converse seed load balancers, info handler functions will provide information about message - size, queuing strategy, pack functio etc	mostly accessed in conv-ldb
chksum			
magic			
padding			