

# MPI Tools Proposals

Jeff Squyres, Cisco Systems

# MPI DLL Locations

---

- ▶ In conjunction with other proposals
  - ▶ Message queue query
  - ▶ MPI handle query (possible)
- ▶ Specify DLL location to debuggers
- ▶ “Old” way was a single compile-time filename string
  - ▶ Not sufficient – what if I have multiple DLLs?
  - ▶ E.g., 32 and 64 bit
- ▶ “New” way: provide an argv-style array of filename strings



# MPI DLL Locations

---

- ▶ **Old way:**

```
char MPIR_dll_name[] = "/path/to/my/dll";
```

- ▶ **New way**

```
char **mpimsgq_dll_locations = NULL;
```

```
int MPI_Init(...) {  
    char **foo = malloc(...);  
    foo[0] = "/path/to/first/dll";  
    foo[1] = "/path/to/next/dll";  
    foo[2] = NULL;  
    mpimsgq_dll_locations = foo;  
}
```

- ▶ **No filename conventions**

- ▶ Values may be filled in at any time during the process

- ▶ Tool examines each filename in the array until it finds a suitable one (if any!)



# MPI Handle Introspection

---

- ▶ Have a tool be able to query an MPI implementation for meta information about MPI handles
  - ▶ Example: hover over MPI\_Comm variable in debugger
    - ▶ Instead of showing its handle value (e.g., 0x239f2b00)
    - ▶ Show “This is <communicator\_name>, <rank> of <size>, an <inter|intra|cartesian|graph> communicator, ...etc.”
    - ▶ Tool can get public information about the back-end object
    - ▶ Can display the info however it wants to (or not)
  - ▶ MPI implementations are free to not implement some or all of this functionality
    - ▶ Query includes “what can you tell me?”
    - ▶ Answer may be “nothing!” (or something very basic, like comm name, rank, size)
-

# MPI Handle Introspection

---

- ▶ Infrastructure similar to message queue interface
- ▶ Sample communicator query functions:

```
int mpidbg_comm_query(mqs_image *image, mqs_image_info *image_info,  
                    mqs_process *process, mqs_process_info *process_info,  
                    mqs_taddr_t comm, struct mpidbg_comm_handle_t **ch)
```

```
int mpidbg_comm_handle_free(struct mpidbg_comm_handle_t *ch)
```



# MPI Handle Introspection

---

- ▶ Infrastructure similar to message queue interface
- ▶ Sample communicator query functions:

```
int mpidbg_comm_query_basic(struct mpidbg_comm_handle_t *ch,  
                           char comm_name[MPIDBG_MAX_OBJECT_NAME],  
                           enum mpidbg_comm_info_bitmap_t *comm_bitflags,  
                           int *comm_rank,  
                           int *comm_size,  
                           int *comm_fortran_handle,  
                           mqs_taddr_t *comm_cxx_handle)
```

```
int mpidbg_comm_query_procs(struct mpidbg_comm_handle_t *ch,  
                            int *comm_num_local_procs,  
                            struct mpidbg_process_t **comm_local_procs,  
                            int *comm_num_remote_procs,  
                            struct mpidbg_process_t **comm_remote_procs)
```

