

Day0

Table of Contents

- [1. Introduction to MPI \(Message Passing Interface\)](#)
- [2. What is MPI?](#)
- [3. History and Development of MPI](#)
- [4. Why MPI?](#)
- [5. MPI Implementations](#)
- [6. Shared vs Distributed Systems](#)
- [7. Downloading and Installing MPI](#)
- [8. Loading MPI on PARAM shavak](#)

1. Introduction to MPI (Message Passing Interface)

2. What is MPI?

MPI stands for Message Passing Interface. It is a standardized and portable message-passing system designed to function on a variety of parallel computing architectures. MPI is widely used for developing parallel applications and is essential for high-performance computing (HPC).

3. History and Development of MPI

MPI was developed in the early 1990s by the MPI Forum, a group consisting of parallel computing vendors, library writers, and application specialists. The need for a standard messaging protocol arose due to the growing diversity of parallel computing environments. Before MPI, parallel programming was fragmented with various proprietary and non-standard systems. MPI was established to provide a unified and efficient way to develop parallel programs.

4. Why MPI?

- **Portability:** MPI is designed to run on various systems, from small clusters to large supercomputers.
- **Performance:** Optimized for high performance on different hardware architectures.
- **Scalability:** Supports efficient communication for large-scale systems.

- **Flexibility:** Offers a wide range of communication operations and can be used in different parallel programming models.

5. MPI Implementations

There are several implementations of MPI, some of the most popular ones include:

- **MPICH:** A high-performance and widely portable implementation of MPI.
- **Open MPI:** An open-source MPI implementation that combines technologies and resources from several projects.
- **MVAPICH:** MPI over InfiniBand, optimized for high performance on InfiniBand clusters.

6. Shared vs Distributed Systems

- **Shared Memory Systems:** Multiple processors access a common memory space. Communication between processors is implicit and achieved through memory read/write operations.
- **Distributed Memory Systems:** Each processor has its own private memory. Communication between processors is explicit and typically achieved through message passing, which is where MPI comes into play.

7. Downloading and Installing MPI

To get started with MPI, you need to download and install an MPI implementation. Here are general steps for downloading and installing Open MPI:

1. **Download Open MPI:** Visit the [Open MPI website](<https://www.open-mpi.org>) and download the latest version of Open MPI.
2. **Extract the tarball:**

```
tar -xvf openmpi-x.y.z.tar.gz
cd openmpi-x.y.z
```

3. **Configure, Build, and Install:**

```
./configure --prefix=/path/to/install
make
make install
```

4. **Set Environment Variables:** Add the following lines to your ``.bashrc`` or ``.bashprofile``:

```
export PATH=/path/to/install/bin:$PATH
export LD_LIBRARY_PATH=/path/to/install/lib:$LD_LIBRARY_PATH
```

8. Loading MPI on PARAM shavak

```
source /home/apps/spack/share/spack/setup.env.sh # source spack package manager
spack find openmpi # check if mpi is installed or not
# spack install -j40 openmpi # if not installed then this command will install
spack load openmpi/_your_hash # load mpi with specific has if multiple version
```

Author: Abhishek Raj

Created: 2024-06-28 Fri 14:09