

Topic: Charm++

Date: April 11, 2024



## **Introduction**

In chare programming, we logically organize objects into chare array classes and modules. The first file created in a chare program is usually the main module. In this class, we will use a single module for simplicity but more complex programs could use a multitude of modules organized into different directories. In the main module, essential components and functions are included to ensure proper execution and error handling.

## **Objects and Arrays in the Main Module**

We create objects in the main chare using constructors that allow for remote calls for object instantiation on different processes. Depending on the program requirements, the structure of arrays can vary as 1D or 2D arrays. The main chore is responsible for coordinating how the program executes and eases the clean exits through functions such as 'done' in the example provided.

## **Handling Main Chare Functions**

In MPI programs, we have seen how mechanisms like `MPI_Finalize` allow for a straightforward cleanup. However, chare programming does not have such a mechanism. To deal with this, the termination is based on task completion signals from other chares that signal when they have finished execution. This synchronization can be done as the shown 'done function'.

## **Proxies and Global Variables**

Proxies ease the communication between chares which allow for remote method invocation. Global variables act like read-only array handles that ensure execution and data exchange by providing information to all the program components.

## **The Hello World Class**

This class is an example of object-oriented programming in chare programming. The class behavior is defined by constructors and entry methods that make interactions between objects smooth. At the same time, parameter marshaling enables efficient data exchange between chares.