Use a namespace for Charm++ code

Charm++ code that is not generated should be in a namespace. Currently name collisions arise (eg Group). In general, Ck should probably be replaced with a namespace. The specific problem case I am encountering is having a group chare derive off of Group where I have defined a Group class in a namespace and want the generated Charm++ code to be in the same namespace. The compiler then searches the namespace first for Group and finds a candidate that does not work.

History

#1 - 09/28/2017 11:12 AM - Nils Deppe
Looking through more of the Charm++ code, this ticket should be made a higher priority than I initially thought. Charm++ is riddled with undefined behavior because of using reserved identifiers (see section 17.6.4.3.2 of the C++11 standard). The standard says "Each name that contains a double underscore _ _ or begins with an underscore followed by an uppercase letter is reserved to the implementation for any use." and "Each name that begins with an underscore is reserved to the implementation for use as a name in the global namespace." The __register function of CkIndex_..., and all the _registerBlah functions just the ones I immediately found.

#2 - 09/28/2017 12:11 PM - Phil Miller
- Target version set to 7 (Next Generation Charm++)

When we take an API break with the next major release, addressing this will be part of it.