proxy - like a pointer / reference

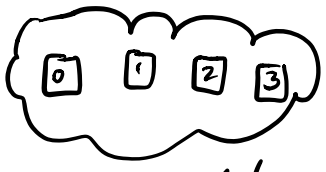
data may not be in your library or processor

charm++ deals with transferring data

Singleton

an imaginary computer w/ a set of instructions

Array (of singleton)




many singletons indexed

1. Parallel components are software

"computing units"

not hardware; can use diff hardware and software will figure out how to distribute

2. more like texting, less like phone

## Actions

task that you can ask parallel components to do;

or elements of an array parallel component

different type of actions.

For pi-dartboard

- throw darts + contribute to reduction
- take reduction results + get pi

## Reduction

a type of talking to each other

1. @ end of action they state their answer
2. add up all answer

a bunch of numbers → one number

a lot of data → aggregate summary

can be an addition, multiplication...

## Meta variables

input file - like options; used when compiling

ex. BBH sim: how many dimensions do you want?

Test_ Algorithm Reduction. cpp

( good template for Pi dartboard

create

Test_ Pi Dart. cpp

Test_PiDart.cpp
- not doing it in parallel yet
- we need an output
- include <random> library
  <iostream>
- throw_darts function
- anonymous namespace will allows you to keep your function exclusive to the file
- print 'value' after the reduction

- each element should use throw_darts func.
- my_send_int{ array_index } → throw_darts
- " .cpp " | c++filt
  clean way to look at errors
- ocean 12 nodes = 240 processors
  (20 processors)
- make parallel components array have 240 elements (1 element / processor)

parallel components are able to communicate w/ eachother

ocean 12 nodes (20 pr / node)
1 processor per node is allocated for communication

cout outputs only 6 sig fig
use <iomanip> to increase numbers we can print out

symbolic link
(ln -s)
ctrl - (to go back)

put in cache - doesn't change
put in databox if maybe change