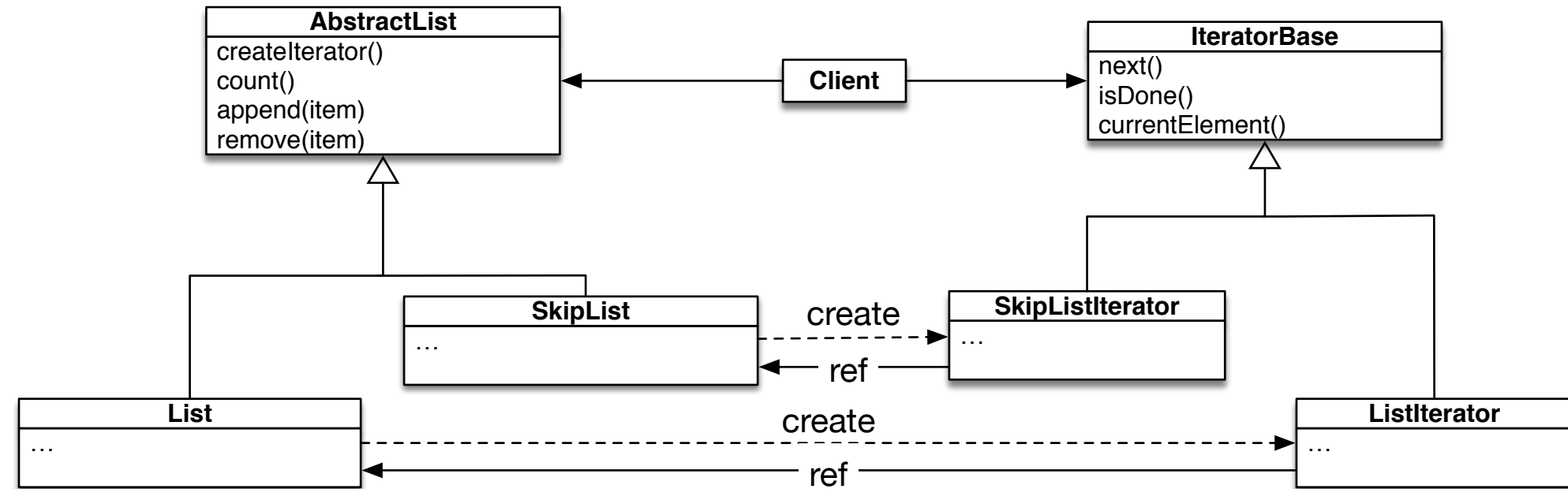


Iterator



Intent: to not bloat container interface with different ways to traverse the container. We might have multiple iterators on the same container at the same time.

Iterator and the container it iterates are coupled: friends of each other. The polymorphic iterator part provides an uniform way to traverse different underlying containers.

Having clients control the iteration offers greater flexibility, iteration algorithm can be defined in the iterator or the container. Having it in the former allows different iteration mechanisms to evolve independently at the cost of less container encapsulation.

A robust iterator ensures adding or removing while traversal won't interfere with traversal, and does it without copying the container. Robust iterators usually require additional mechanisms in the container, e.g. registering the iterator with the container.