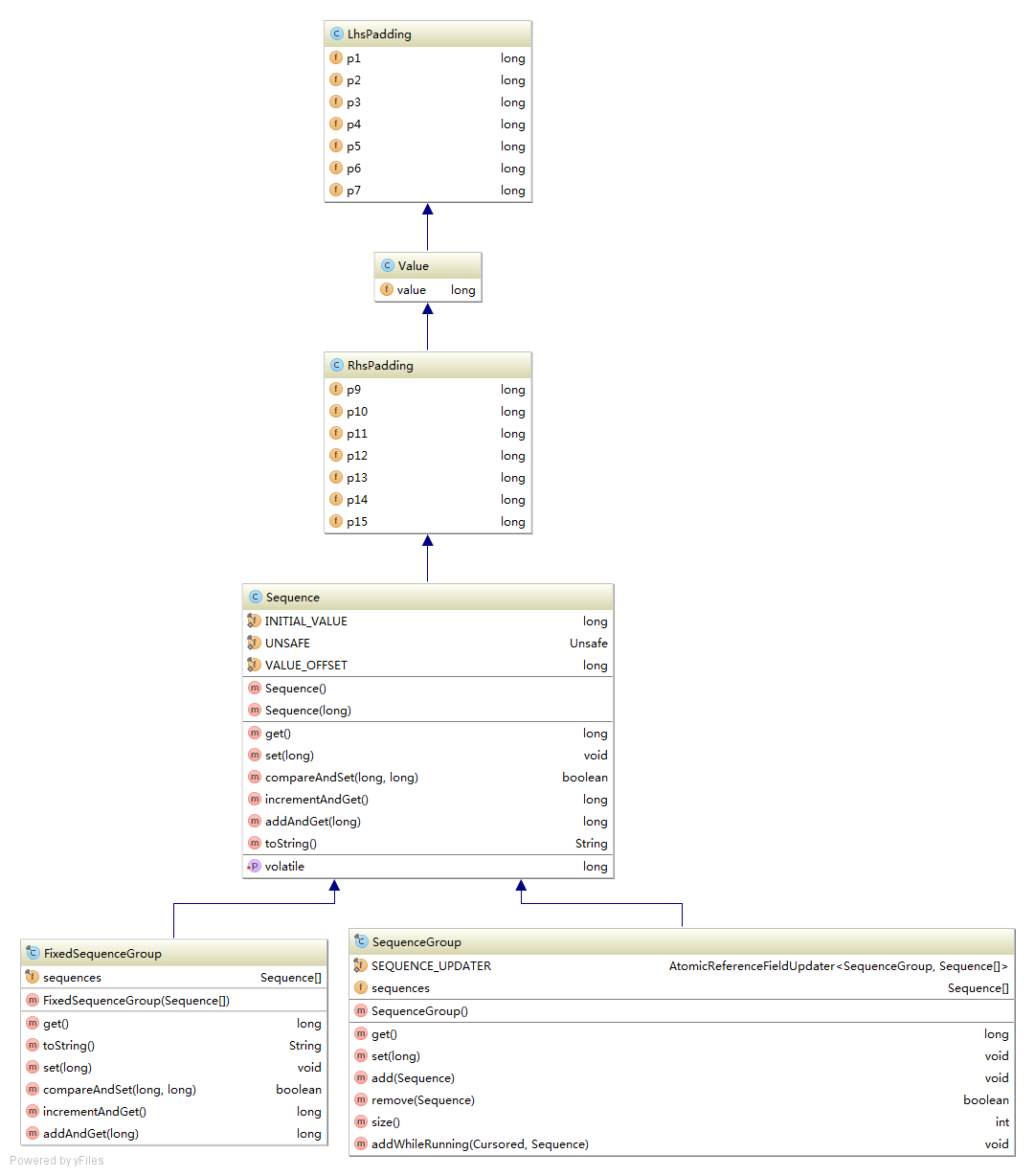
**Disruptor中的序列-Sequence**



Disruptor中较为重要的一个类是Sequence。我们设想下，在disruptor运行过程中，事件发布者(生产者)和事件处理者(消费者)在ringbuffer上相互追逐，由什么来标记它们的相对位置呢？它们根据什么从ringbuffer上发布或者处理事件呢？就是这个Sequence-序列。

Sequence是Disruptor最核心的组件。其通过顺序递增的序号来编号管理通过其进行交换的数据（事件），对数据(事件)的处理过程总是沿着序号逐个递增处理。一个Sequence用于跟踪标识某个特定的事件处理者(RingBuffer/Consumer)的处理进度。生产者对RingBuffer的互斥访问，生产者与消费者之间的协调以及消费者之间的协调，都是通过Sequence实现。几乎每一个重要的组件都包含Sequence。由于需要在线程间共享，所以Sequence是引用传递，并且是线程安全的；再次，Sequence支持CAS操作；最后，为了提高效率，Sequence通过padding来避免伪共享。

总结一下：

1. 通过Sequence的一系列的继承关系可以看到，它真正的用来计数的域是value，在value的前后各有7个long型的填充值，这些值在这里的作用是做cpu cache line填充，防止发生伪共享；
2. value域本身由volatile修饰，而且又看到了Unsafe类，大概猜到是要做原子操作了；
3. Sequence是一个做了缓存行填充优化的原子序列；