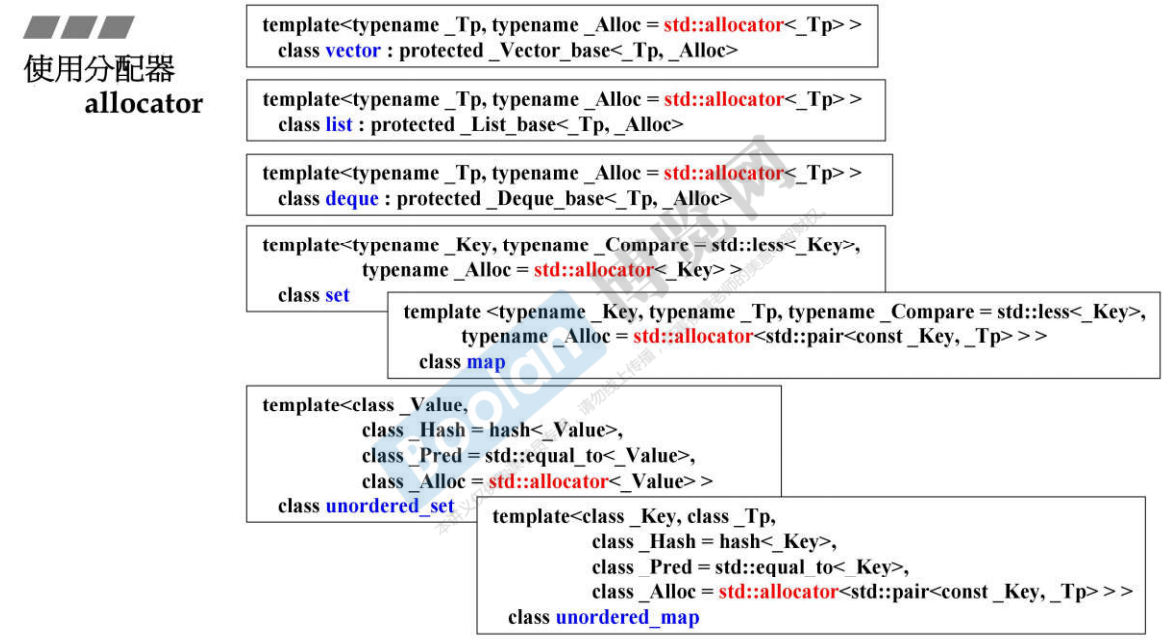
1、



容器的背后需要分配器的支持它对内存的使用。

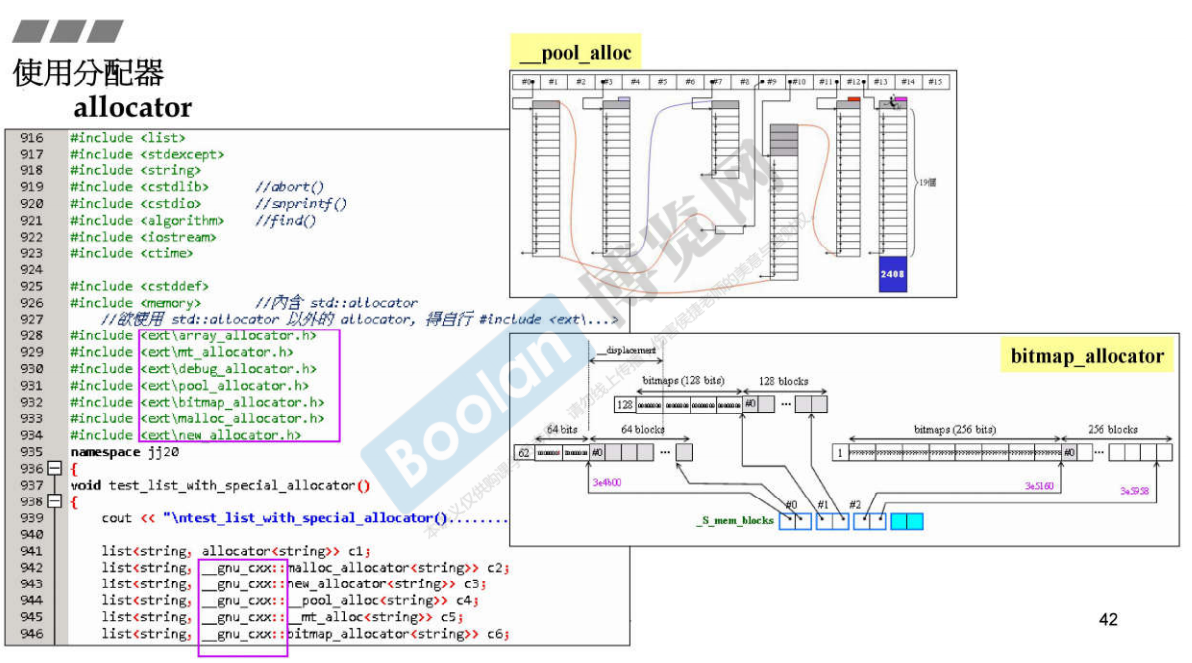
分配器在这里一定是存在的东西，但对于程序员来说最好是透明的。（容器默认有一个分配器）

2、

要知道使用的容器对应的是什么分配器

每一个容器都是一个模板（毋庸置疑的），然后对于每一个模板参数列表中，第二个参数是默认的

3、



这里只能使用GNU编译器编译（绿色头文件圈住部分）

这里的mt是多线程的意思，multi\_thread

这里的pool是内存池

这里的ext文件夹下的意思是扩充性的，非标准的

这些源代码被定义在\_\_gnu\_cxx这里，而非std

4、



先选择一个容器choice，都是list，每一个容器用的分配器不一样

然后输入elements元素个数

5、

我们其实可以直接使用分配器，分配器是一个class，有allocate和deallocate

但这没有必要，因为我们真正使用的是容器，分配器只是为容器服务

直接使用分配器是上面右图

这里 p = alloc1.allocate(1)是分配一个int

6、

在容器之外我们需要分配内存时当然使用new和delete，或者malloc和free

例如用malloc时我们需要指明多少字节才能拿到多少内存的指针，free释放时不需要指明，这样的做法才好才方便，这样只需要直到开辟内存时开发者需要多少就开多少，而释放时开发者不一定记得开辟时拿了多少内存。

而deallocate在返回内存时不仅要指明返还的指针还要指明还多少内存。

如果指明返还的没有开辟时那么多，后果老师不知道。