the strong typed nature of C++ and especially U++ leaves you with a lot of control and responsibility. imho constructing object without care (like it is in C#) is both easy and wasting performance, inviting for careless handling with ressources. knowing what's up and needed is best practice to keep apps reactive. beeing that said, i go along with mr_pen & mirek.

this does not mean that Shared<> is just bad per se. it is merley a helper. but seeing it as an invitation to go GC is a bit too far i think. as mirek said, GC is compiler support. either have it or dont. there is no safe way in between.

some more info about shared_ptr usage, which clearly shows that it brings a lot of hassle as well. hassle one can spare when investing the thinking power in a correct model. a lot of work to save work is cumbersome.
  http://www.codeproject.com/KB/stl/boostsmartptr.aspx#Example : Using shared_ptr in containers