Subject: Hopefully final C++11 related disruptive change: Callback now deprecated
Posted by mirek on Tue, 04 Oct 2016 08:28:31 GMT
View Forum Message <> Reply to Message

After numerous changes to better use C++11 features in U++, I was still a little bit uneasy about Function/Callback duality that was originally forced by overloading issues. Now after further investigation it looks like it is actually possible to deprecate Callback completely and replace everywhere with Function, with only slight impact on backward compatibility.

So the new model says there is single Upp::Function, similar to std::function (with two very important distinctions - empty Function call is NOP and it can be combined using <<) and convenience typedefs of Function to Event and Gate. Callback is replaced by Event<> (which is equivalent of Function<void ()>, Callback1<Foo> by Event<Foo> (Function<void (Foo)>) etc, Gate is similar to original Gate (e.g. Gate<int> is now Function<bool (int)>). Callback and related things (like THISBACK) are kept for BW compatibility.

In my 2 major project, conversion to this new model required no changes in one of them and fixing 5 lines in another. Things to fix are reported as compile time bugs. There are basically two situations to fix:

```
Ctrl x;
x <<= x.WhenSomething = THISBACK(Foo);
```

has to be rewritten as

```
x.WhenAction = x.WhenSomething = THISBACK(Foo);
```

(note that x <<= THISBACK(Foo) is OK).

```
Gate foo;
```

has to be rewritten as

```
Gate<> foo;
```

(however this second issue is very rare).

Hopefully this was the last thing to change in U++ before "C++11 embracing release"...

Mirek