Subject: Problem with Vector::Add (pick/clone semantics)
Posted by shutalker on Fri, 09 Aug 2019 10:14:12 GMT
View Forum Message <> Reply to Message

Hi all!

I've encountered with the following problem. When I try to define such object as

```
const static VectorMap<String, Vector<String>> MY_MAP = {
    {"s1", pick(Vector<String>{"s11", "s12", "s13", "s14"})},
    {"s2", pick(Vector<String>{"s21", "s22", "s23", "s24"})},
    {"s3", pick(Vector<String>{"s31", "s32", "s33", "s34"})},
    {"s4", pick(Vector<String>{"s41", "s42", "s43", "s44"})},
    {"s5", pick(Vector<String>{"s51", "s52", "s53", "s54"})}
};
```

I get several errors like this

```
/home/alexis/upp/uppsrc/Core/Vcont.hpp (158): error: call to implicitly-deleted copy constructor of
'Upp::Vector<Upp::String>'
...
/upp/uppsrc/Core/Core.h (357): In file included from /home/alexis/upp/uppsrc/Core/Core.h:357:
 (): T *q = new(Rdd()) T(x);
/home/alexis/upp/uppsrc/Core/Vcont.h (132): note: in instantiation of member function
'Upp::Vector<Upp::Vector<Upp::String> >::GrowAdd' requested here
 (): T&      Add(const T& x)               { return items < alloc ? *(new(Rdd()) T(clone(x))) :
GrowAdd(x); }
/home/alexis/upp/uppsrc/Core/Map.h (51): note: in instantiation of member function
'Upp::Vector<Upp::Vector<Upp::String> >::Add' requested here
 (): T&      Add(const K& k, const T& x)          { key.Add(k); return value.Add(x); }
/home/alexis/upp/uppsrc/Core/Map.h (179): note: in instantiation of member function
'Upp::AMap<Upp::String, Upp::Vector<Upp::String>, Upp::Vector<Upp::Vector<Upp::String> >
>::Add' requeste
d here
 (): AMap(std::initializer_list<std::pair<K, T>> init) { for(const auto& i : init) Add(i.first, i.second); }
/home/alexis/upp/uppsrc/Core/Map.h (236): note: in instantiation of member function
'Upp::AMap<Upp::String, Upp::Vector<Upp::String>, Upp::Vector<Upp::Vector<Upp::String> >
>::AMap' reques
ted here
 (): VectorMap(std::initializer_list<std::pair<K, T>> init) : B::AMap(init) {}
```

I guess the reason is

```
T *q = new(Rdd()) T(x); // <-- should be clone(x)
```

So I made a little patch that fixed the problem. Please, check it and give feedback if I did

something wrong

UPD
I use upp from git repository https://github.com/ultimatepp/mirror

Used compiler: FreeBSD clang version 6.0.0 (tags/RELEASE_600/final 326565) (based on LLVM 6.0.0)

File Attachments

1) vcont.patch, downloaded 373 times