
Subject: Re: Versioning support (hack-ishly solved)
Posted by [slashupp](#) on Mon, 12 Aug 2019 06:00:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

Added history-keeping to the header/macro...

The new header is:

```
#ifndef _VersionRBT_h_
#define _VersionRBT_h_

#define V_RELEASE 0
#define V_BUILD 0
#define V_TEST 0
#define N_2_S0(x) #x
#define N_2_S(x) N_2_S0(x)
#define V_VERSION "v " N_2_S(V_RELEASE) "." N_2_S(V_BUILD) "." N_2_S(V_TEST)
```

```
//=====
====
```

```
//history will be updated by the Versioning-macro..
```

```
// and can be accessed with something like:
```

```
// {
// std::string s{"History\n"};
// for (auto p:VersionHistory<>) { s+="  ["; s+=p.first; s+="] -> "; s+=p.second; s+="\n"; }
// PromptOK(DeQtf(s.c_str()));
// }
```

```
#include <map>
```

```
template<typename S=std::string> struct MVHIST
{
    std::map<S, S> m;
    MVHIST(const S &k, const S &v) { m[k]=v; }
    MVHIST<S>& operator()(const S &k, const S &v) { m[k]=v; return *this; }
    operator std::map<S, S>() { return m; }
};
```

```
template<typename S=std::string> std::map<S, S> VersionHistory=MVHIST<S>
//HISTORY (insertion-point, leave this comment in place - see macro)
;
```

```
#endif
```

and the macro is attached.

File Attachments

1) [Versioning.usc](#), downloaded 285 times
