
Subject: Re: is there a way to print debug output?
Posted by [mirek](#) on Sat, 28 Apr 2007 05:53:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

Core/Diag.h

```
#define LOG(a)      UPP::LockLog(), UPP::VppLog() << a << '\n', UPP::UnlockLog()
#define LOGF      UPP::__LOGF__
#define LOGBEGIN() UPP::LockLog(), UPP::VppLog().Begin()
#define LOGEND()   UPP::VppLog().End(), UPP::UnlockLog()
#define LOGBLOCK(n) RLOGBLOCK(n)
#define LOGHEXDUMP(s, a) UPP::HexDump(VppLog(), s, a)
#define QUOTE(a)    { LOG(#a); a; }
#define LOGSRCPOS() UPP::LockLog(), UPP::VppLog() << __FILE__ << '#' << __LINE__ <<
'\n', UPP::UnlockLog()
#define DUMP(a)     UPP::LockLog(), UPP::VppLog() << #a << " = " << (a) << '\n',
UPP::UnlockLog()
#define DUMPC(c)    UPP::LockLog(), UPP::DumpContainer(VppLog() << #c << ":\n", (c)),
UPP::UnlockLog()
#define DUMPC2(c)   UPP::LockLog(), UPP::DumpContainer2(VppLog() << #c << ":\n", (c)),
UPP::UnlockLog()
#define DUMPC3(c)   UPP::LockLog(), UPP::DumpContainer3(VppLog() << #c << ":\n", (c)),
UPP::UnlockLog()
#define XASSERT(c, d) if(!bool(c)) { LOG("XASSERT failed"); LOGSRCPOS(); LOG(d);
ASSERT(0); } else
#define NEVER()     ASSERT(0)
#define XNEVER(d)   if(1) { LOG("NEVER failed"); LOGSRCPOS(); LOG(d); ASSERT(0); }
else
#define CHECK(c)    if(!bool(c)) { ASSERT(0); } else
#define XCHECK(c, d) if(!bool(c)) { LOG("XCHECK failed"); LOGSRCPOS(); LOG(d);
ASSERT(0); } else

#define TIMING(x)   RTIMING(x)
#define HITCOUNT(x) RHITCOUNT(x)
#define ACTIVATE_TIMING() UPP::TimingInspector::Activate(true);
#define DEACTIVATE_TIMING() UPP::TimingInspector::Activate(false);

#define DLOG(x)     LOG(x)
#define DDUMP(x)    DUMP(x)
#define DDUMPC(x)   DUMPC(x)
```

Result is in (app_name).log file in the output directory (in Win32) or .(app_name) (in posix environments).

Mirek
