

## messagefacility - Bug #17457

### Message fields not being filled in

08/14/2017 03:54 PM - Eric Flumerfelt

<b>Status:</b>	Closed	<b>Start date:</b>	08/14/2017
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Christopher Green	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	8.00 hours
<b>Target version:</b>	2.09.00	<b>Spent time:</b>	8.00 hours
<b>Description</b>			
<p>I'm using MessageFacility v2_00_02, and trying to debug its interaction with the Message Facility Viewer in mf-extensions. I have noticed that the following fields are not being filled in messages from artdaq:</p> <p>hostname hostaddr application PID</p> <p>From what I can tell, these were filled in in the ErrorLog.cc file, which has been removed from the repository (most likely due to code restructuring). The find . -type f xargs grep -i setHostname command shows that the setters for these fields are still defined, but never called:</p> <p>v2_00_02: [eflumerf@ironwork messagefacility]\$ find . -type f xargs grep -i setHostname ./Utilities/ErrorObj.cc: void mf::ErrorObj::setHostName(std::string const&amp; hostname) ./Utilities/ErrorObj.cc: myXid.setHostName(hostname); ./Utilities/MessageFacilityMsg.cc: void MessageFacilityMsg::setHostName(std::string const&amp; hostname) ./Utilities/MessageFacilityMsg.cc: ep-&gt;setHostName(hostname); ./Utilities/ELextendedID.h: void setHostName(std::string const &amp; hostname); ./Utilities/ELextendedID.h: mf::ELextendedID::setHostName(std::string const &amp; hostname) ./Utilities/ErrorObj.h: virtual void setHostName ( const std::string &amp; hostname ); ./Utilities/MessageFacilityMsg.h: void setHostName (std::string const&amp; hostname);</p> <p>v1_18_04: [eflumerf@ironwork messagefacility]\$ find . -type f xargs grep -i setHostName ./MessageService/ErrorLog.cc: a-&gt;msg().setHostName(a-&gt;hostname()); ./MessageService/ErrorLog.cc: if ( updateHostName ) msg.setHostName ( a-&gt;hostname() ); ./MessageService/ErrorLog.cc: if ( updateHostName ) msg.setHostName ( "" ); ./Utilities/ErrorObj.cc: void ErrorObj::setHostName(std::string const&amp; hostname) ./Utilities/MessageFacilityMsg.cc: void MessageFacilityMsg::setHostName(std::string const&amp; hostname) ./Utilities/MessageFacilityMsg.cc: ep-&gt;setHostName(hostname); ./Utilities/ErrorObj.h: virtual void setHostName ( const std::string &amp; hostname ); ./Utilities/MessageFacilityMsg.h: void setHostName (std::string const&amp; hostname);</p> <p>Please advise. Thanks, Eric</p>			

#### Associated revisions

**Revision f459abc6 - 08/28/2017 06:45 PM - Chris Green**

Fix and regression test for issue #17457.

#### History

**#1 - 08/16/2017 04:30 PM - Christopher Green**

- Tracker changed from Bug to Support
- Status changed from New to Feedback
- Assignee set to Christopher Green

Hi Eric,

I'm pretty sure this was discussed as part of our meeting and associated correspondence back in March: it was explained (I think) that ErrorLog was just convenience functions for a likely set of calls to the context updating routines, which, as you note, still exist. I seem to remember, although my memory is very sketchy at the best of times, that the advice was to put calls to the individual context update routines where you need them -- I don't even think ErrorLog is something you want to appropriate for yourselves (although that would be perfectly fine).

These four items you mention are only useful in an inter-process / inter-node context, and as such we believe it is appropriate for artdaq code to set the context where appropriate.

If you have trouble working out where to plug in the appropriate calls, or there are other issues, we'd be happy to get together and discuss them. I'll put the issue into feedback mode to facilitate that; otherwise, please let us know how you proceeded so we can resolve the ticket.

**#2 - 08/21/2017 12:01 PM - Kyle Knoepfel**

- *Tracker changed from Support to Bug*
- *Status changed from Feedback to Assigned*
- *Estimated time set to 8.00 h*

We will add interface to allow you to set this information.

**#3 - 08/28/2017 06:46 PM - Christopher Green**

- *Status changed from Assigned to Resolved*

**#4 - 08/28/2017 06:58 PM - Christopher Green**

- *Target version set to 2.09.00*

Resolved with commit [f459abc](#).

The default application name, hostname, hostaddr and pid have been restored to the message.

In addition to `mf::MessageLoggerQ::setApplication()`, one may now invoke `mf::MessageLoggerQ::setHostName()`, `mf::MessageLoggerQ::setHostAddr()` or `mf::MessageLoggerQ::setPID()` anytime after the call to `StartMessageFacility` (e.g. the constructor of any destination).

N.B.: if the application (or anyone else) calls any of the above functions directly or indirectly (like the free function `mf::SetApplicationName`, for example), then calls from a destination's constructor will be overridden as these happen as a result of the call to `StartMessageFacility()`. To avoid this, your application should call `StartMessageFacility` with an optional second `std::string` argument indicating the initial application name. Art relies on the default setting of the application name, so this should not be an issue.

See the test code at [source:messagefacility/test/Integration/Issue17457TestDestination\\_mfPlugin.cc](#) for an example of calling these functions from a destination's constructor, although since the old defaults are now set and honored once more this may not be necessary for you.

Note also that any of the items in an xid may be overridden in the destination on a per-message basis if you require.

**#5 - 08/28/2017 06:58 PM - Christopher Green**

- *% Done changed from 0 to 100*

**#6 - 10/18/2017 01:47 PM - Kyle Knoepfel**

- *Status changed from Resolved to Closed*