



# NVMe\* Compliance Test Resources

David Woolf, University of New Hampshire InterOperability Laboratory

Sarika Mehta, Validation Lead, Intel Corporation

- ***IOL INTERACT tool***
  - ***David Woolf – UNH-IOL***
- **NVMe Compliance Test Suite**
  - **Sarika Mehta – Intel**

1. Overview of NVMe Test Program and IOL INTERACT Test Tool
2. Requirements for using IOL INTERACT
3. Installation and Use of IOL INTERACT

# 1. Overview of NVMe\* Test Program

- Since 2012 NVMe Promoters Group collaborating with UNH-IOL to create a Test Program for NVMe.



University of New Hampshire  
**InterOperability  
Laboratory**

# 1. Overview of NVMe\* Test Program

## ■ Program has 4 Key Elements

### ■ Test Services

- <https://www.iol.unh.edu/services/testing/NVMe/>



### ■ Plugfests

- <https://www.iol.unh.edu/services/testing/NVMe/plugfestfaq.php>



### ■ *Test Tools*

- <https://www.iol.unh.edu/services/testing/NVMe/tools.php>



### ■ Integrators List

- <https://www.iol.unh.edu/services/testing/NVMe/integratorslist.php>



# 1. Overview of NVMe\* Test Program

-  NVMe Promoters Group
  - Provides guidance on services, tools, plugfests, Integrators List qualification requirements
-  UNH-IOL
  - Executes test services and plugfests, administers Integrators List, and develops test tools



# 1. Overview of NVMe\* Test Program

- Key Test Tool Developed by UNH-IOL is **IOL INTERACT**
- Purpose of **IOL INTERACT**
  - Quickly and easily verify that an SSD meets the conformance requirements for the NVMe Integrators List
  - Not a rigorous performance test
- Sits on top of **tNVMe** compliance software



## 2. Requirements for using IOL INTERACT NVMe\* Test Software

- tNVMe
  - Installed automatically by IOL INTERACT installer
- PC with Intel Chipset, running Ubuntu 10.10 64 bit.
  - Support for more modern OS to be addressed through tNVMe updates

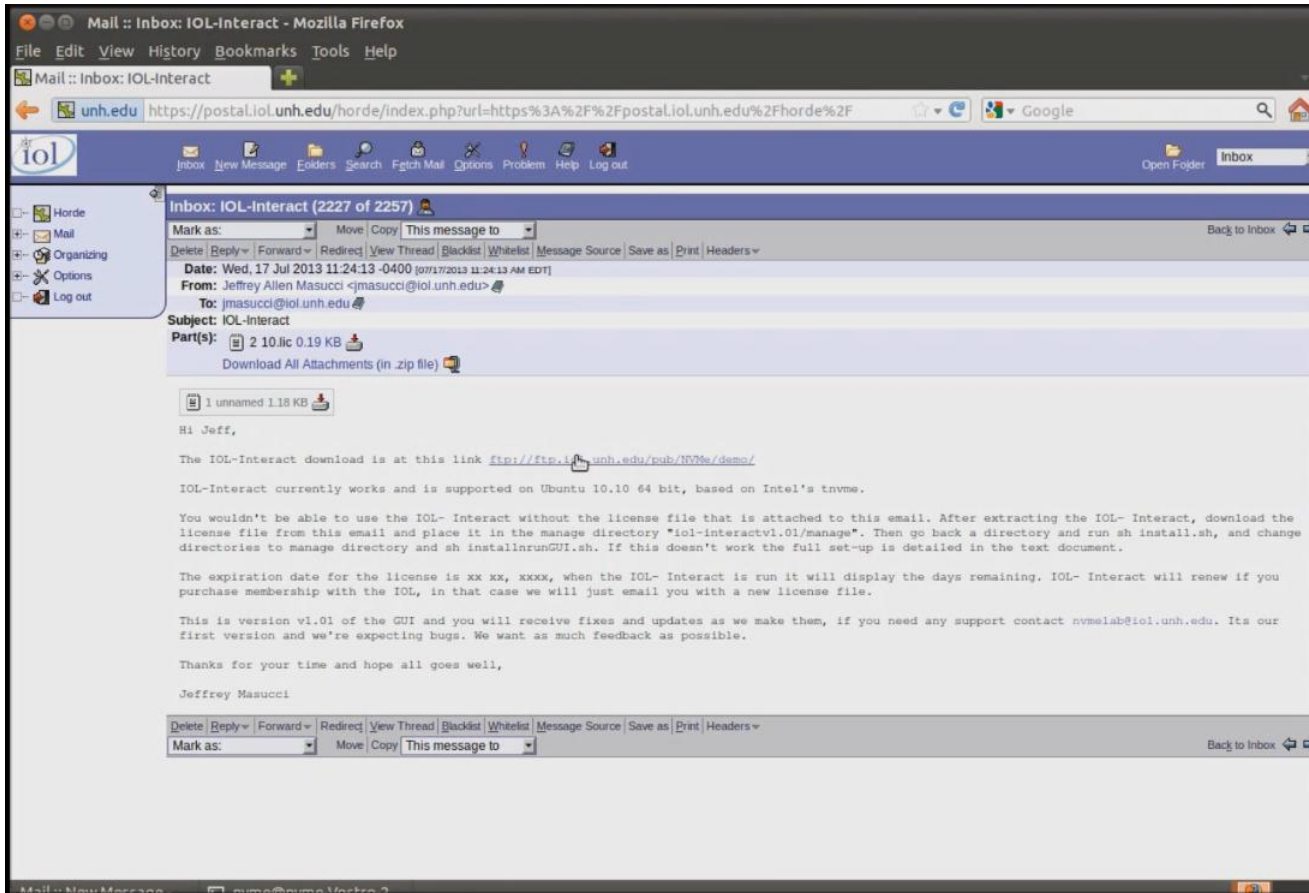


## 2. Requirements for using IOL INTERACT NVMe\* Test Software

- PCIe storage device
  - Any form factor that can connect to the Host System (CEM, SFF-8639 etc...)
- Membership in UNH-IOL NVMe Consortium
  - Grants license for IOL INTERACT, allows you to pre-check compliance for NVMe IL
  - UNH-IOL offers NVMe and PCIe conformance and interop testing
  - <https://www.iol.unh.edu/services/membership/form/>

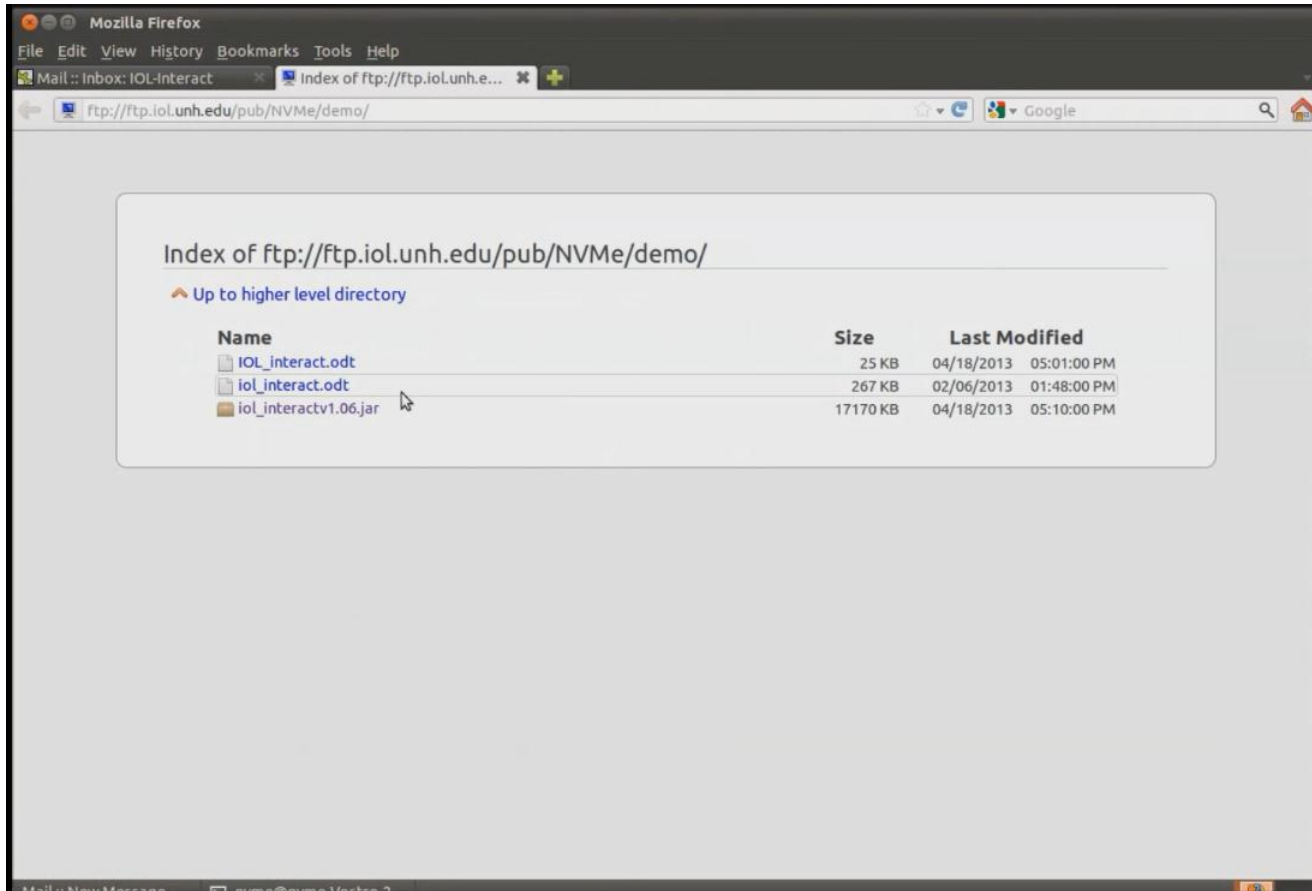
# 3. Installation and Use of IOL INTERACT NVMe\* Test Software

Receive email with IOL INTERACT download link



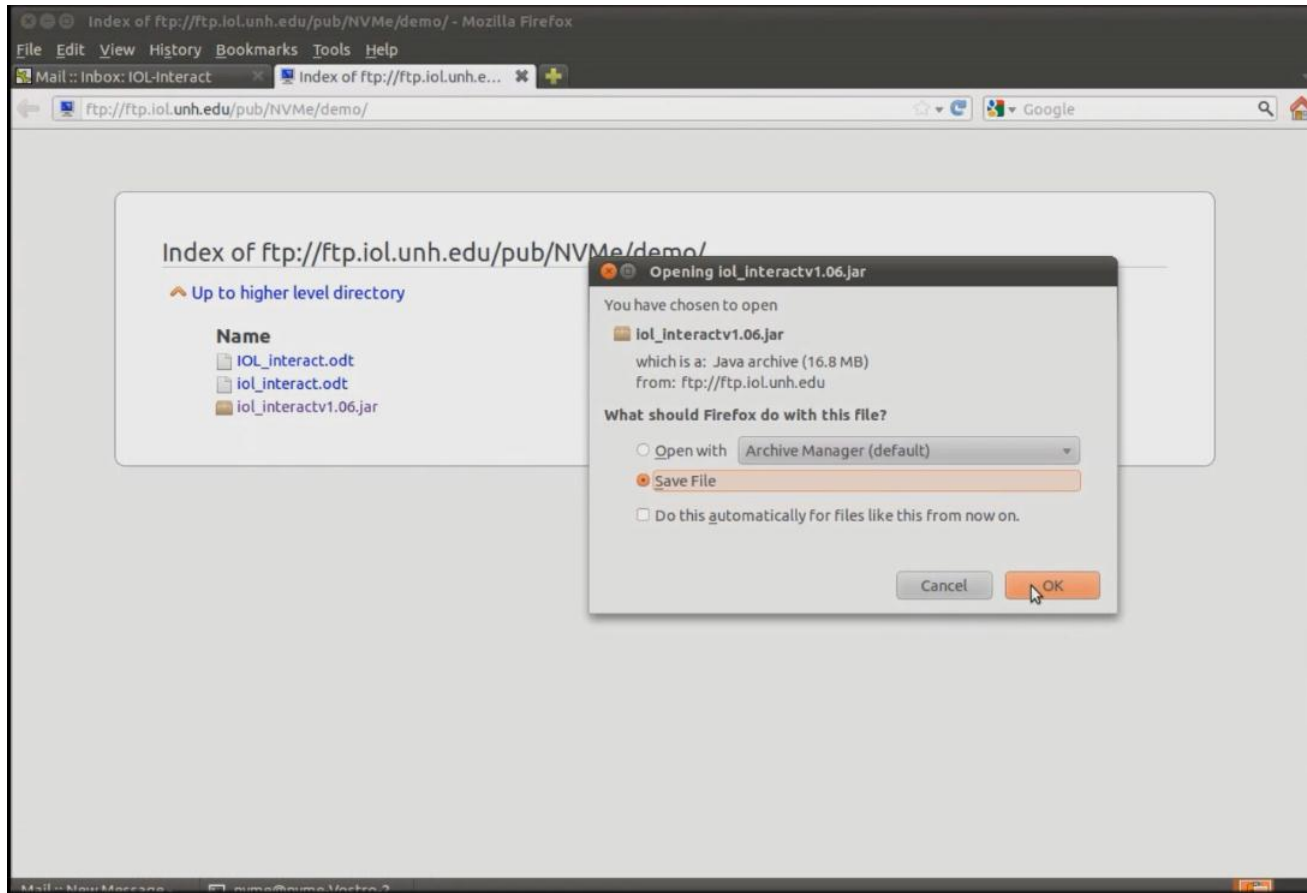
# 3. Installation and Use of IOL INTERACT NVMe\* Test Software

Download the  
jar file



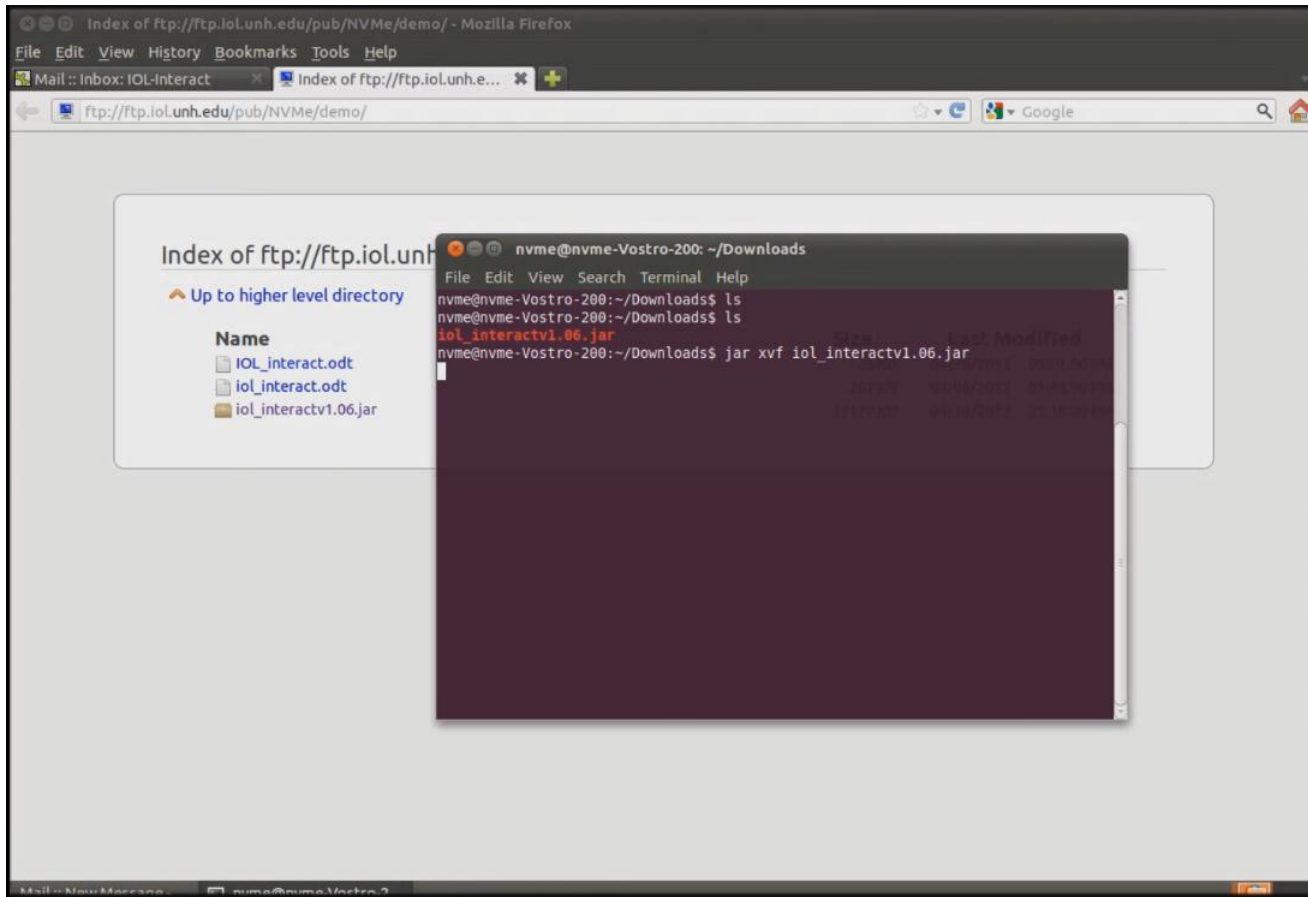
# 3. Installation and Use of IOL INTERACT NVMe\* Test Software

Save it locally



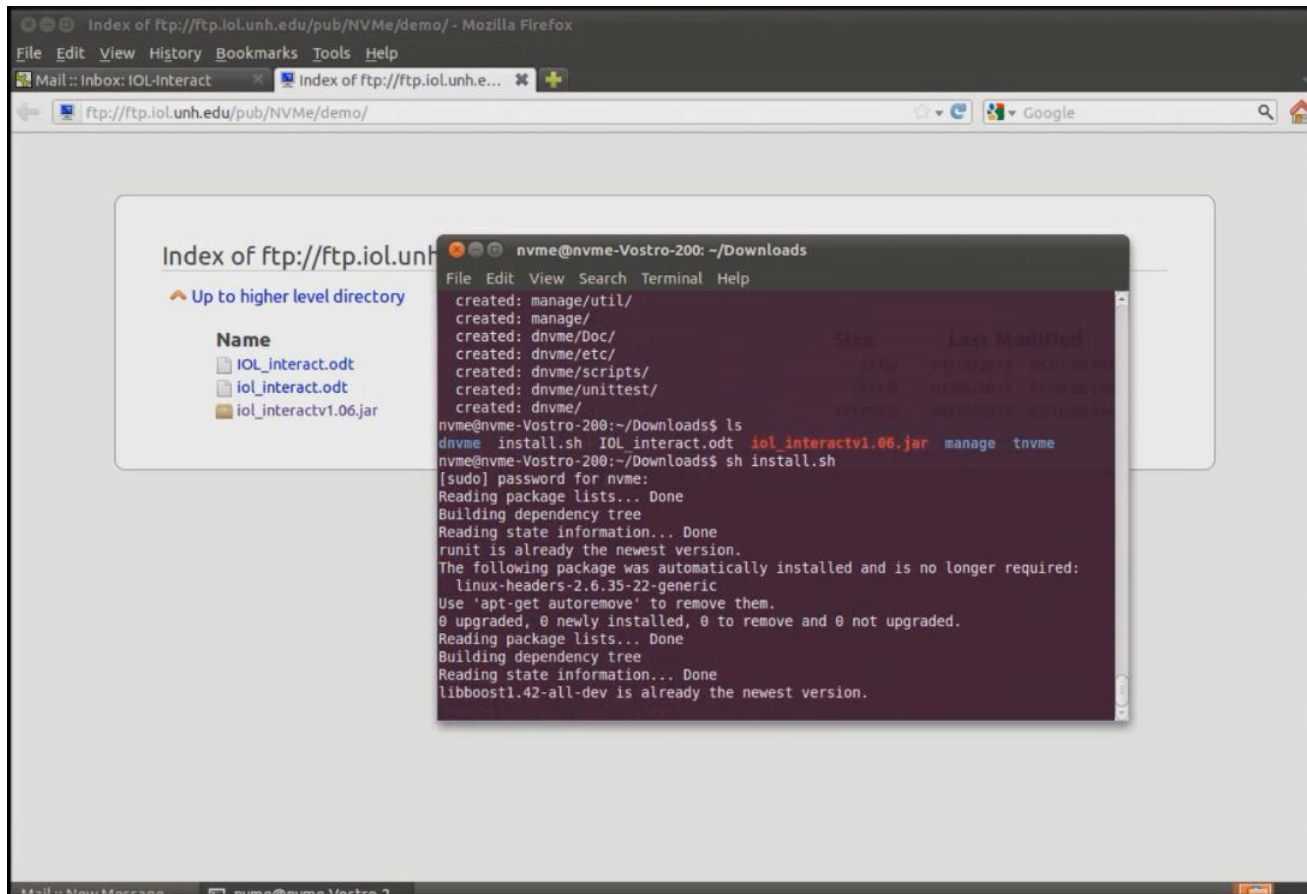
## 3. Installation and Use of IOL INTERACT NVMe\* Test Software

Extract using  
'jar xvf'

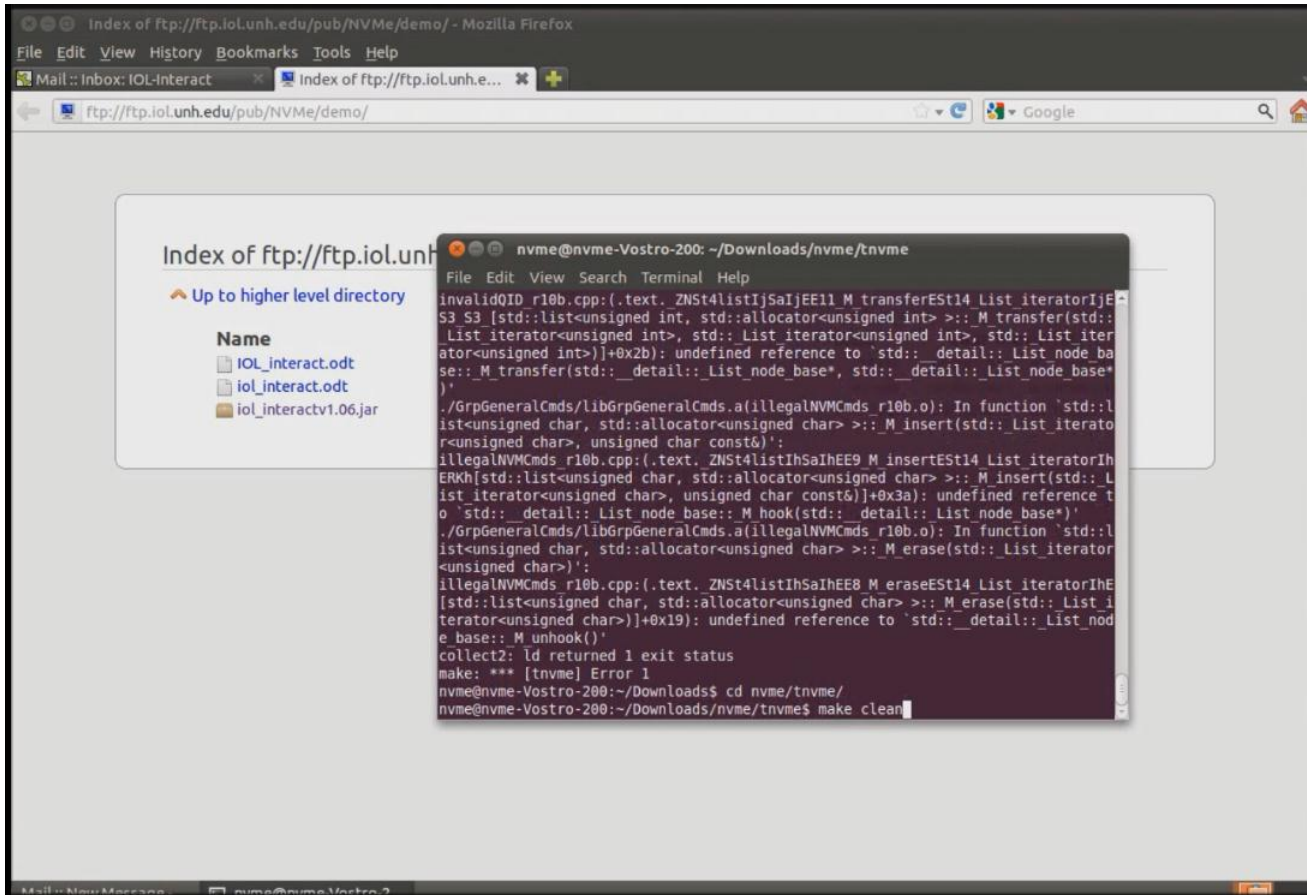


# 3. Installation and Use of IOL INTERACT NVMe\* Test Software

'sh install.sh'



## 3. Installation and Use of IOL INTERACT NVMe\* Test Software

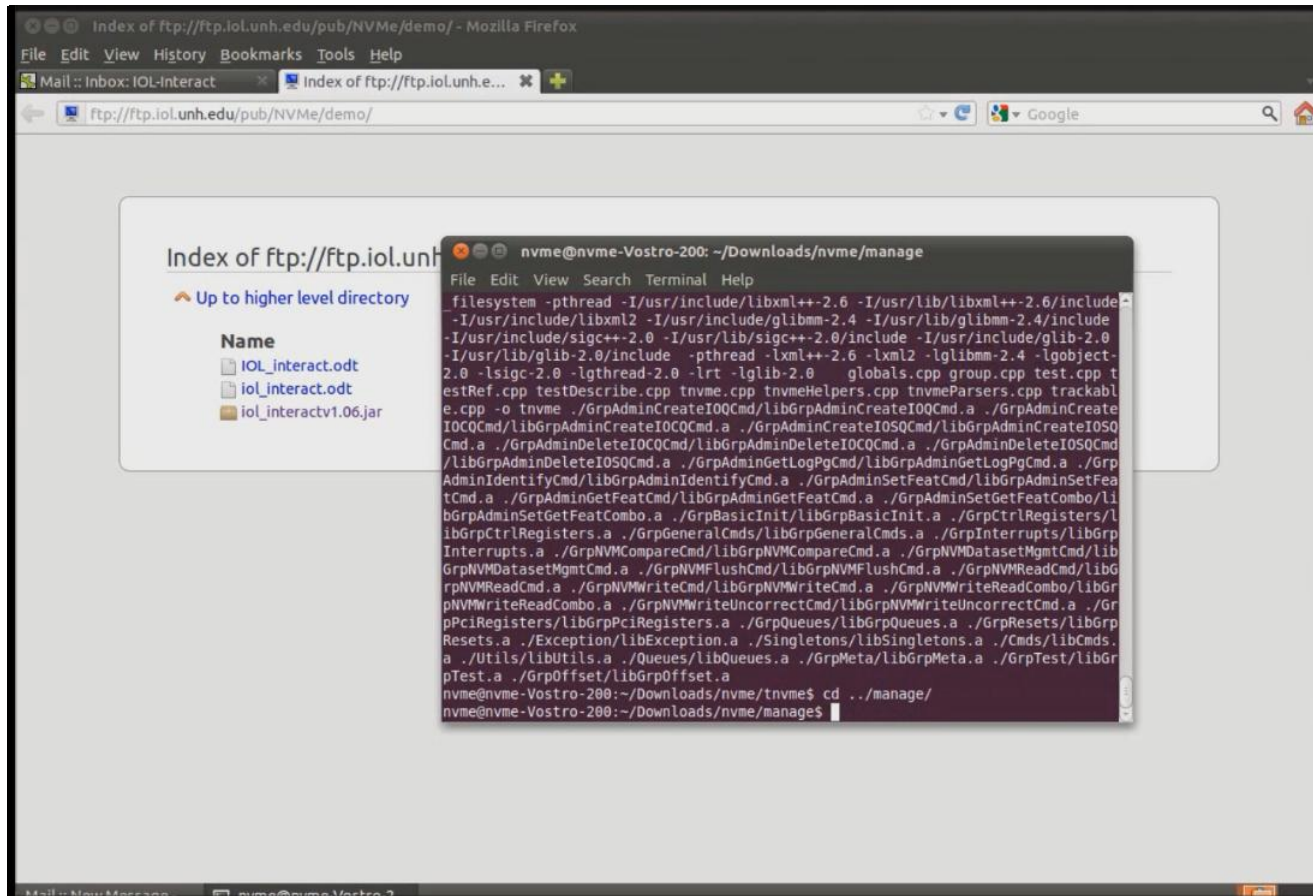


```
nvme@nvme-Vostro-200: ~/Downloads/nvme/tnvme
File Edit View Search Terminal Help
invalidOID_r10b.cpp:(.text.ZNSt4listIjSaIjEE11 M_transferEst14 List_iteratorIjE
S3_S3 [std::list<unsigned int, std::allocator<unsigned int> >::M_transfer(std::
_list_iterator<unsigned int>, std::_List_iterator<unsigned int>, std::_List iter
ator<unsigned int>)+0x2b): undefined reference to `std::detail::List_node ba
se::M_transfer(std::detail::List_node_base*, std::detail::List_node_base*
)'.
./GrpGeneralCmds/libGrpGeneralCmds.a(illegalNVMeCmds_r10b.o): In function `std::l
ist<unsigned char, std::allocator<unsigned char> >::_M_insert(std::_List_iterato
r<unsigned char>, unsigned char const&)':
illegalNVMeCmds_r10b.cpp:(.text.ZNSt4listIhSaIhEE9 M_insertEst14 List_iteratorIh
ERKh[std::list<unsigned char, std::allocator<unsigned char> >::_M_insert(std::_L
ist_iterator<unsigned char>, unsigned char const&)+0x3a): undefined reference to
`std::detail::List_node_base::M_hook(std::detail::List_node_base*)'.
./GrpGeneralCmds/libGrpGeneralCmds.a(illegalNVMeCmds_r10b.o): In function `std::l
ist<unsigned char, std::allocator<unsigned char> >::_M_erase(std::_List_iterator
<unsigned char>)':
illegalNVMeCmds_r10b.cpp:(.text.ZNSt4listIhSaIhEE8 M_eraseEst14 List_iteratorIhE
[std::list<unsigned char, std::allocator<unsigned char> >::_M_erase(std::_List i
terator<unsigned char>)+0x19): undefined reference to `std::detail::List_node
_base::M_unhook()'
collect2: ld returned 1 exit status
make: *** [tnvme] Error 1
nvme@nvme-Vostro-200:~/Downloads$ cd nvme/tnvme/
nvme@nvme-Vostro-200:~/Downloads/nvme/tnvme$ make clean
```

If a make error occurs for tnvme, run 'make clean' followed by 'make'. This will take a minute or so

# 3. Installation and Use of IOL INTERACT NVMe\* Test Software

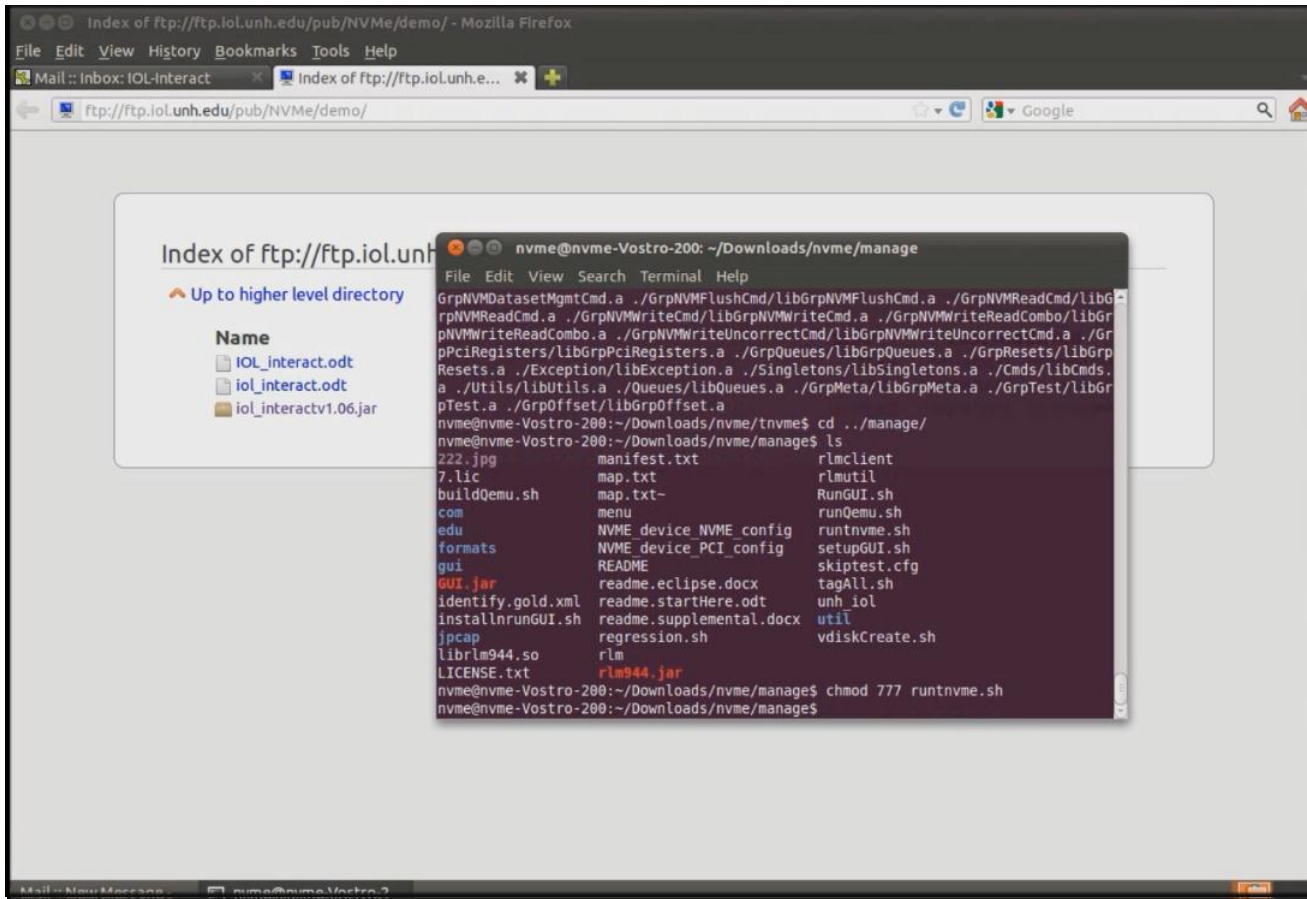
Change directories to /nvme/manage



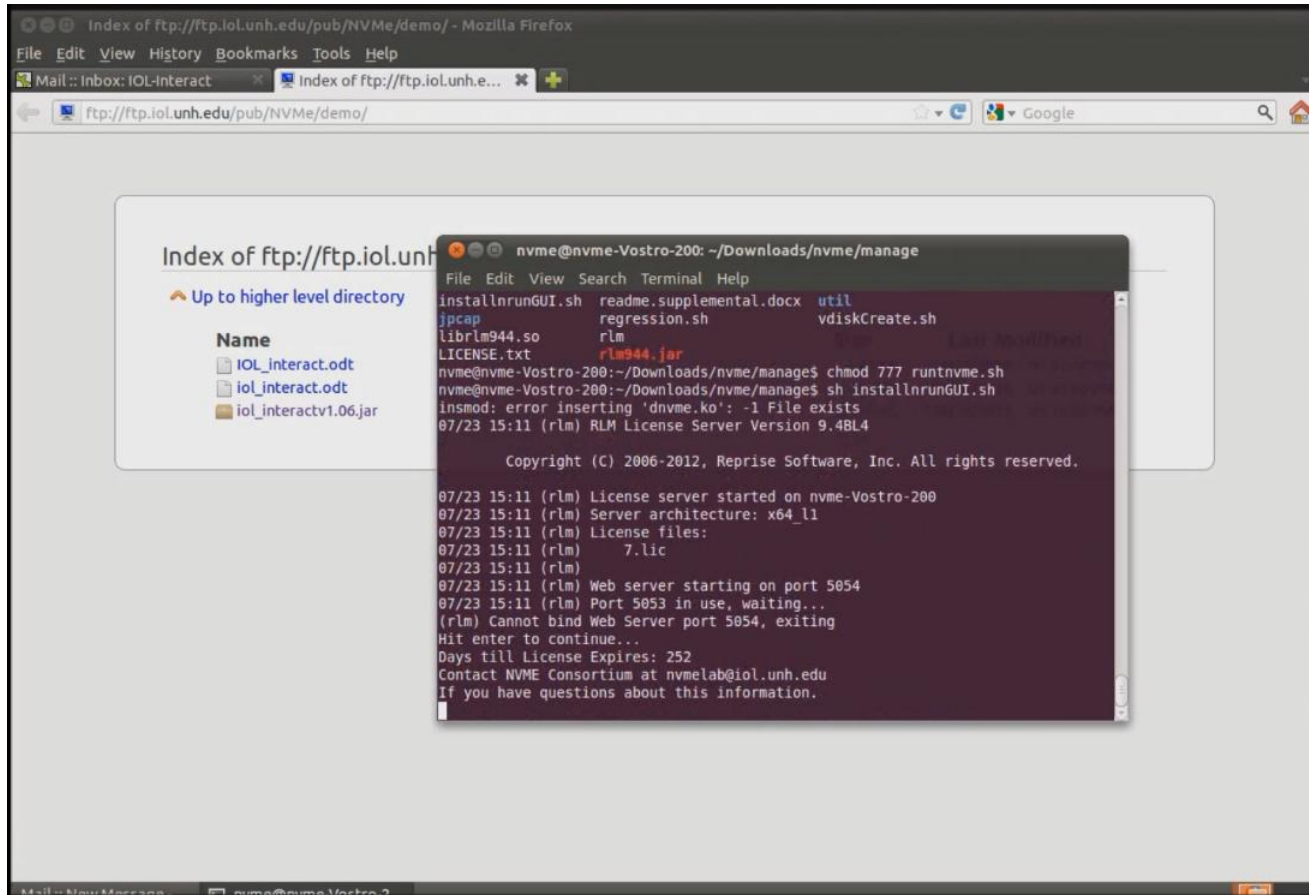


# 3. Installation and Use of IOL INTERACT NVMe\* Test Software

Change permissions for `runtnvme.sh` with `'chmod 777 runtnvme.sh'`



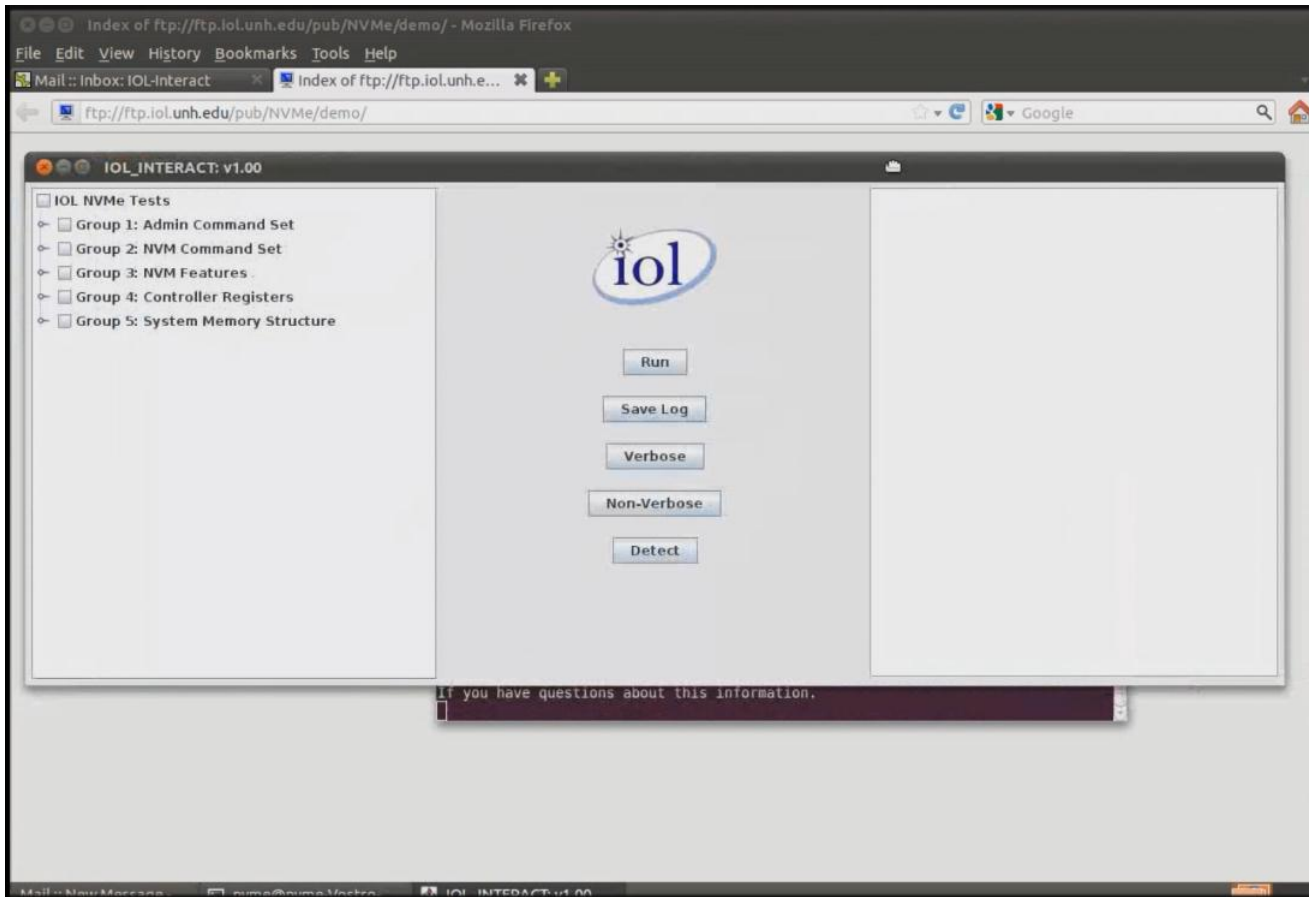
# 3. Installation and Use of IOL INTERACT NVMe\* Test Software



Start IOL  
INTERACT  
with 'sh  
installnrunG  
UI.sh'

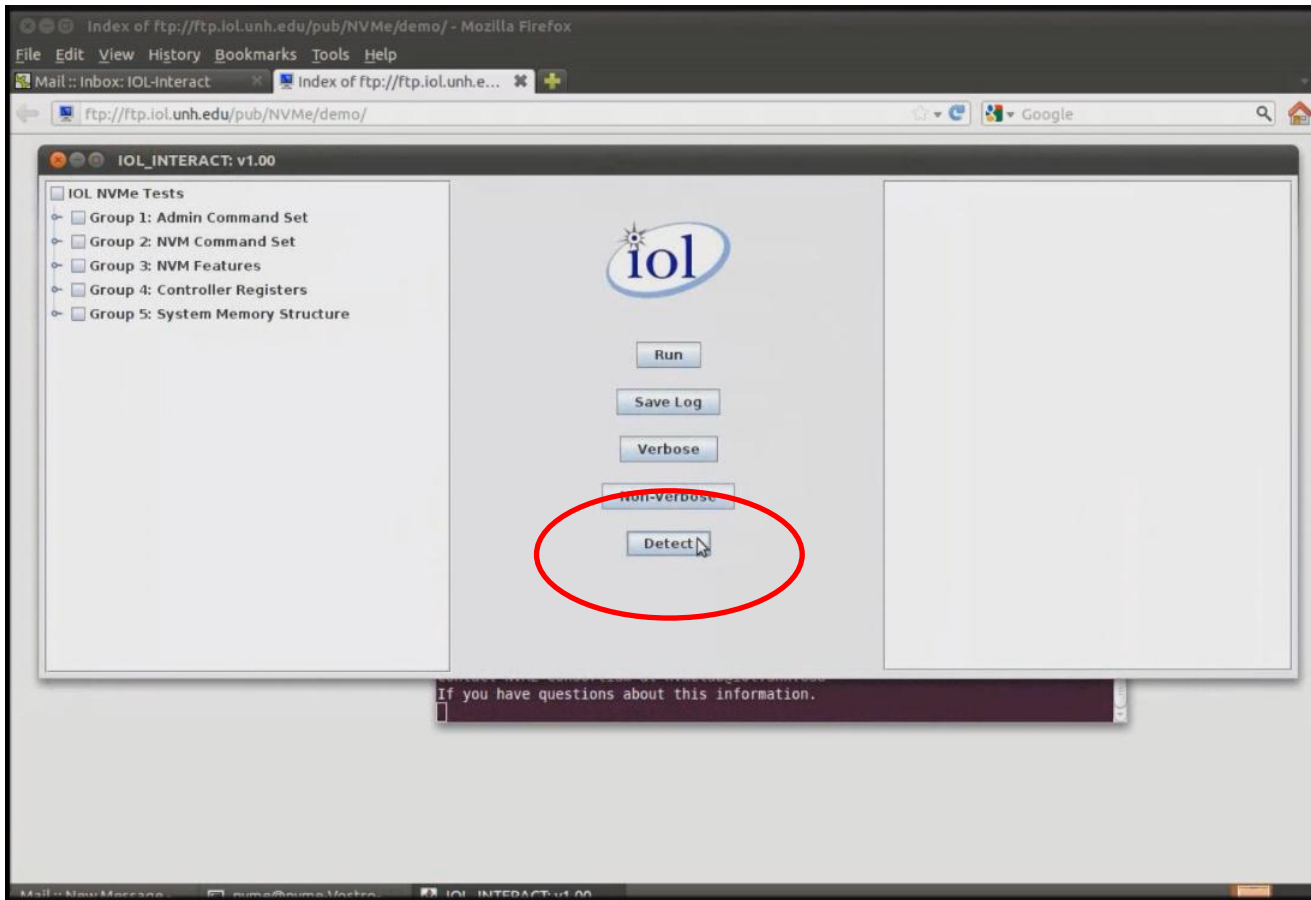
You will see a  
license  
message  
and a UNH-  
IOL contact  
email

# 3. Installation and Use of IOL INTERACT NVMe\* Test Software

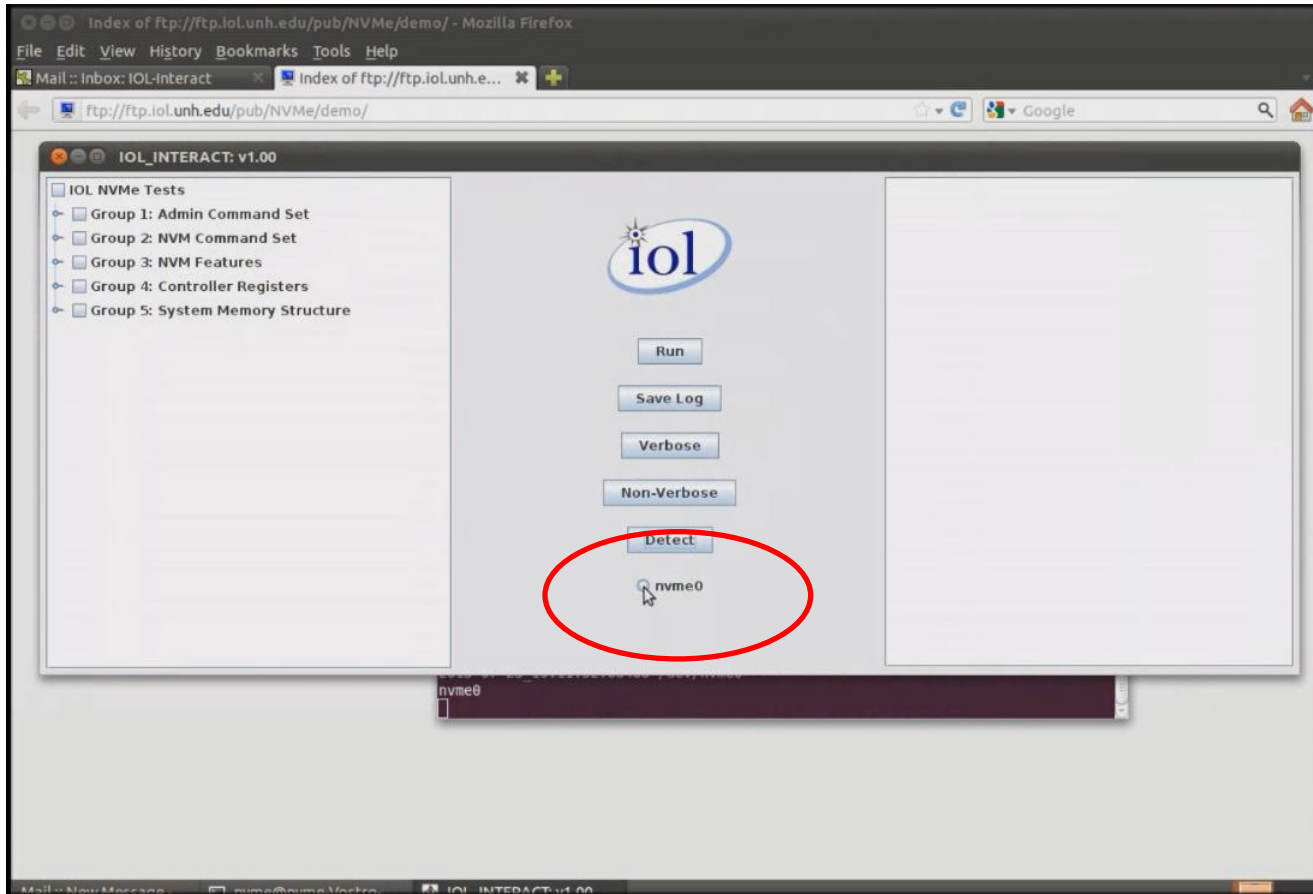


## 3. Installation and Use of IOL INTERACT NVMe\* Test Software

The software will open. Use the 'detect' button to detect any attached NVMe devices.



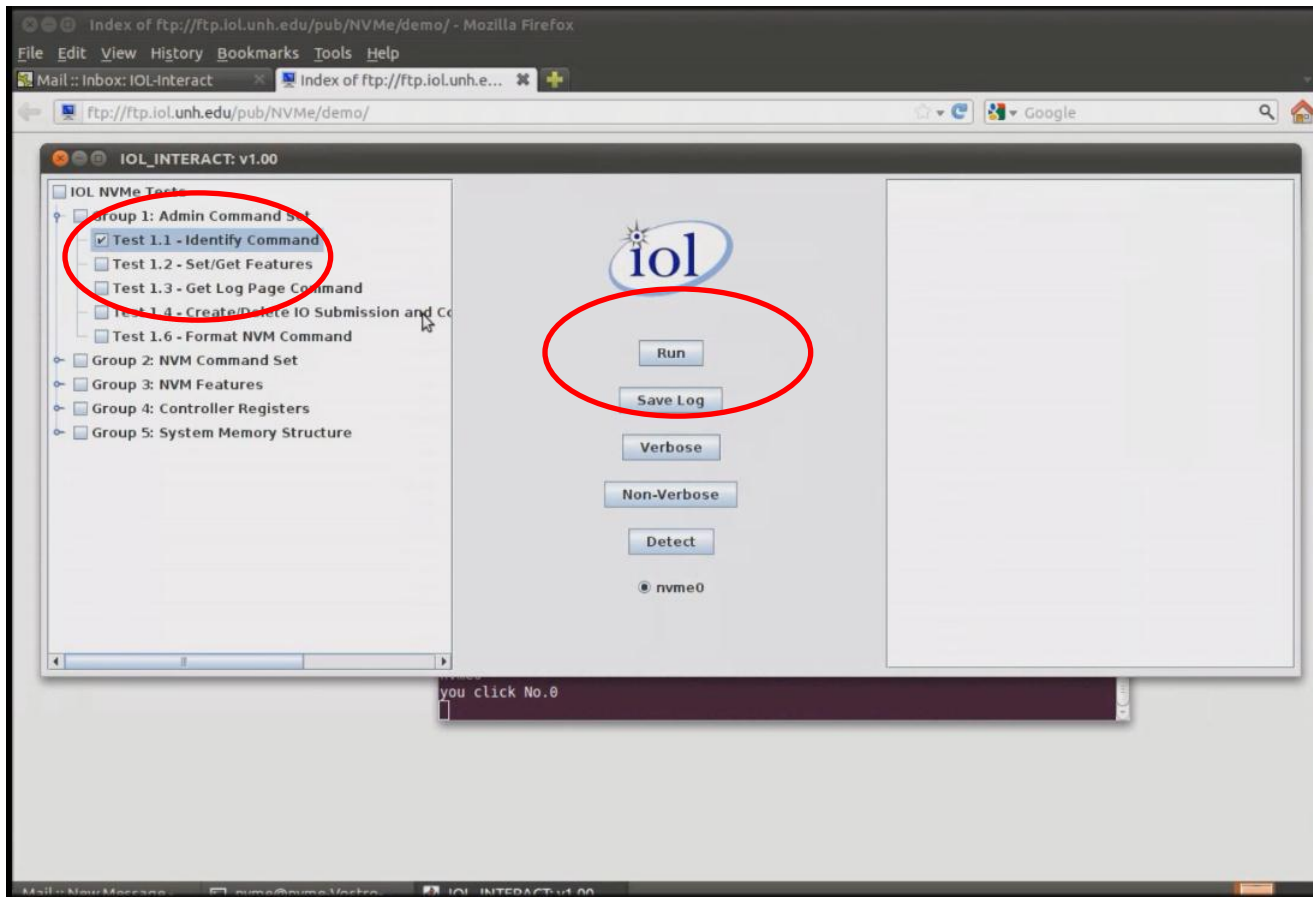
## 3. Installation and Use of IOL INTERACT NVMe\* Test Software



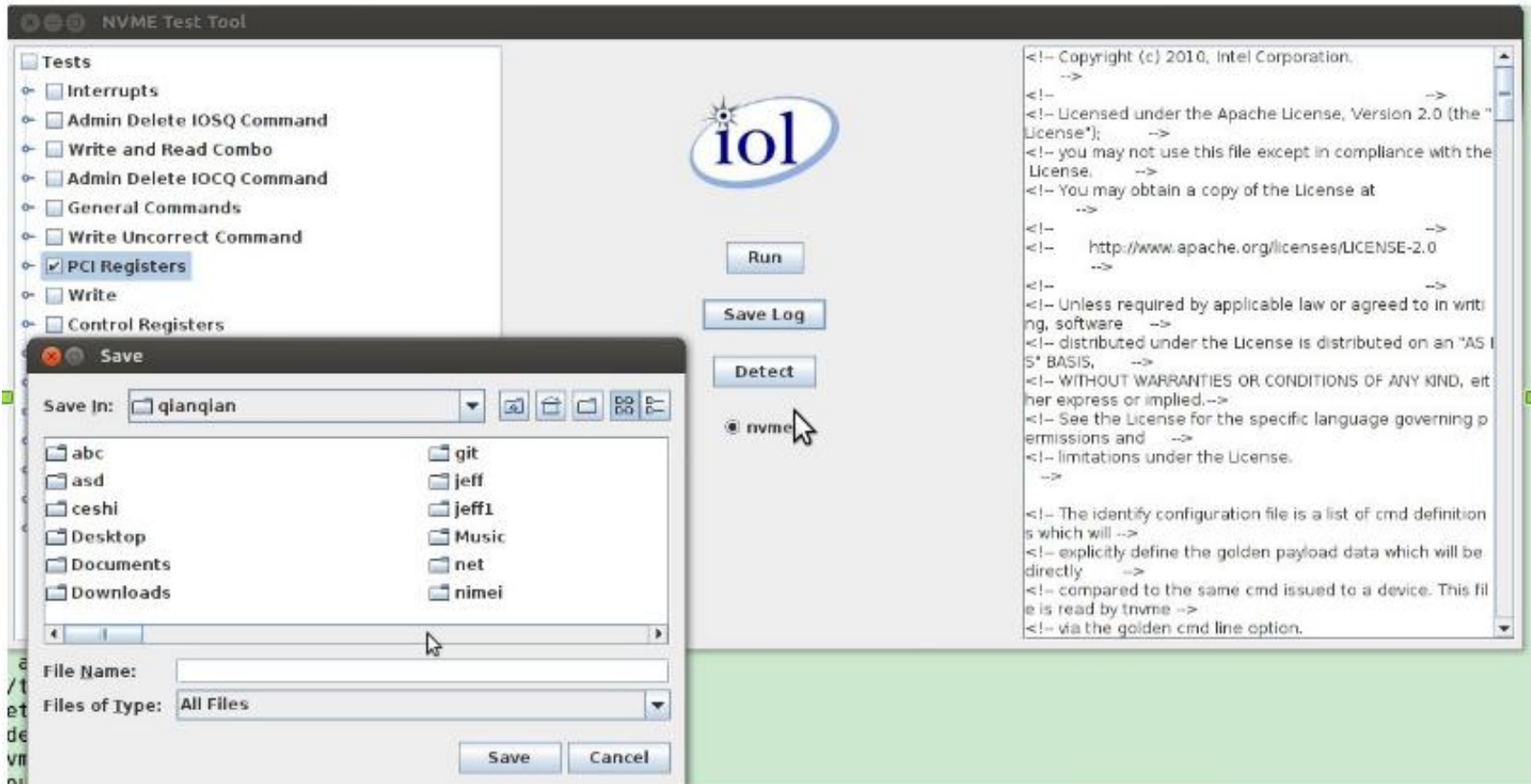
Select the radio button for NVMe device to test. The device must appear in /mnt/nvme

## 3. Installation and Use of IOL INTERACT NVMe\* Test Software

Select tests to run on the left, then click Run.



# 3. Installation and Use of IOL INTERACT NVMe\* Test Software



- IOL INTERACT software sits on top of NVMe compliance test suite software.
- NVMe compliance test suite to be discussed next.



- IOL INTERACT\* tool
  - David Woolf – UNH-IOL\*
- ***NVMe Compliance Test Suite***
  - ***Sarika Mehta – Intel***



# NVMe\* Compliance Test Suite

## Overview

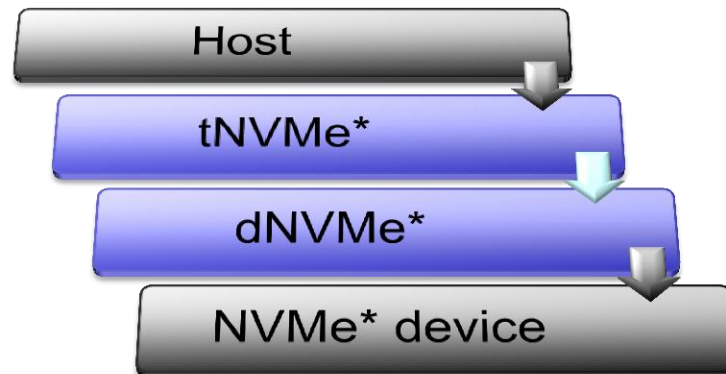
- High-level Introduction
- Usage
- Community Participation

## Introduction

- Open source
- 64-bit Linux\* based, user space software test infrastructure
- Helps validate device compliance with the NVMe\* specification
- Currently supports NVMe\* spec revision 1.0

## Test Suite Components

- tNVMe\* - Compliance test suite application
  - User level interface
  - Weaves functionality provided by kernel into tests
- dNVMe\* – Linux driver for the compliance test suite
  - Exposes an interface to kernel functionality



## Hardware for Test Execution

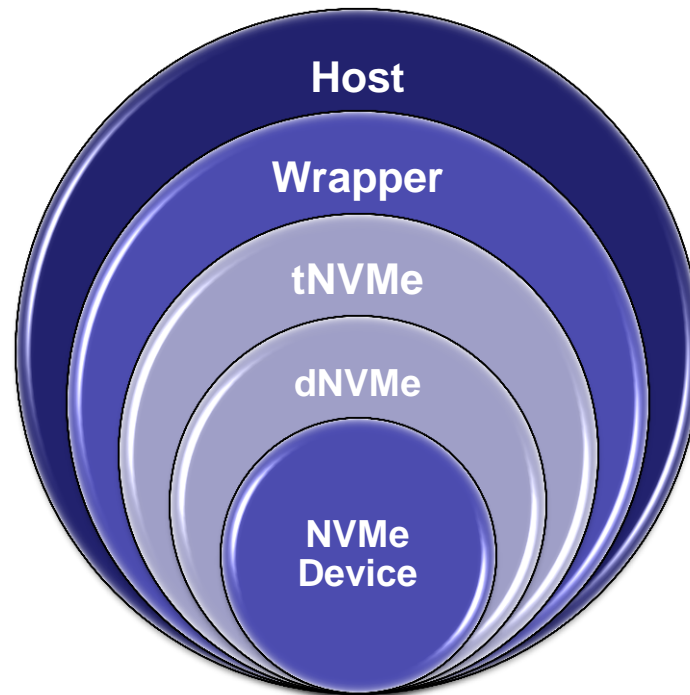
- Real physical hardware
- Simulated hardware

## Test execution with ease of debug

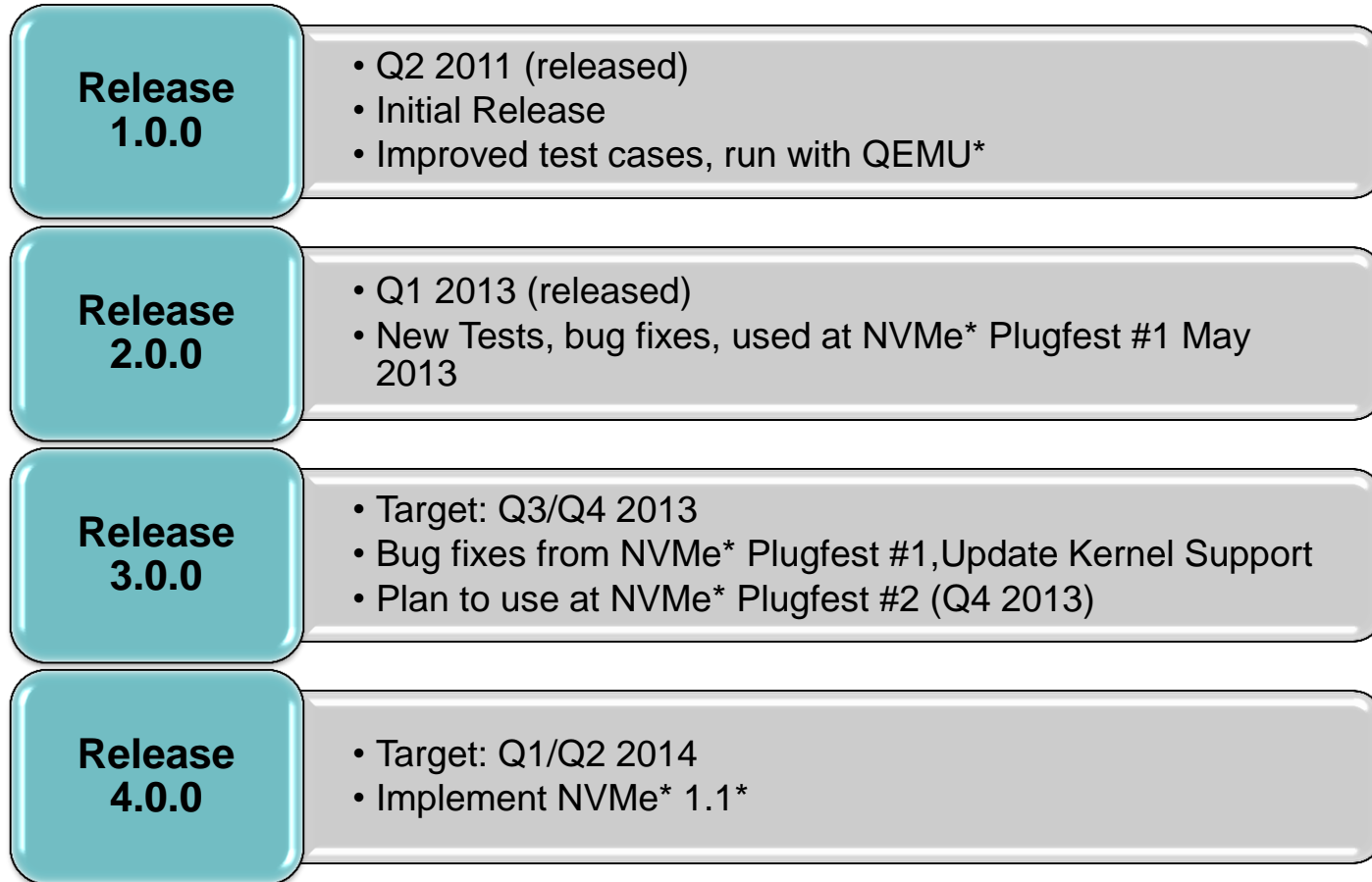
- Zero Dependency
  - Standalone tests
- Configuration Dependency
  - Test depending on prior configuration
- Sequence Dependency
  - Multiple tests depending on prior tests

# NVMe\* Compliance Test Suite

- Use tNVMe\* as is with quick scripting
- Custom debug/validation framework wrapper



# NVMe Compliance Test Suite Planned Roadmap



## Call to Action - Participate

- UNH-IOL\* administers code contributions via Github\*
- Software release: <https://github.com/nvmecompliance>
- Architecture, compiling, executing, and contribution details: <https://github.com/nvmecompliance/tnvme/wiki>
- NVMe\* device simulation: <https://github.com/nvmeqemu>
- Contact: [compliance@nvmexpress.org](mailto:compliance@nvmexpress.org)



## Links and Resources

1. tNVMe\* wiki: <https://github.com/nvmecompliance/tnvme/wiki>
2. Software release: <https://github.com/nvmecompliance>
3. Test plan and test cases:  
<https://github.com/nvmecompliance/tnvme/blob/master/Doc/testPlan.ods>.
4. Designing test dependencies:  
<https://github.com/nvmecompliance/tnvme/blob/master/Doc/testDependencyPr eso.pdf>
5. Code sample for wrapper:  
<https://github.com/nvmecompliance/manage/blob/master/regression.sh>
6. Support: [compliance@nvmexpress.org](mailto:compliance@nvmexpress.org)



# Legal Notices and Disclaimers

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL® PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. INTEL PRODUCTS ARE NOT INTENDED FOR USE IN MEDICAL, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS.

Intel, and the Intel logo are trademarks of Intel Corporation in the United States and other countries.

\* Other names and brands may be claimed as the property of others.

Copyright © 2013 Intel Corporation.