Skyline and msconvert in Docker: access powerful MS data processing tools beyond Windows environments

Matt Chambers[1]; Brendan MacLean[1]; Michael J. MacCoss[1]
[1]University of Washington, Seattle, WA

Introduction

Skyline[1] and msconvert[2] are widely used in the mass spectrometry (MS) fields of proteomics and metabolomics. Skyline is primarily an interactive application that deals with targeted quantification while msconvert converts between MS data formats with minimal user interaction. Both programs use the ProteoWizard software library to read directly from vendor proprietary data formats (e.g. Thermo RAW, Agilent MassHunter). Vendor file reading support is a significant part of Skyline's popularity as most similar tools require an intermediate conversion step to mzML, which takes time and disk space.

Problem

For most users, running Skyline and msconvert was until recently limited to computers running Microsoft Windows because these tools depend on vendor software libraries which could only run there. However, for high throughput analyses, users often want to run on high performance computing (HPC) clusters which almost always run some version of Linux. As a result, tool developers have been trying to get msconvert's vendor readers to run on Linux for many years.

Solution

Wine is a software suite that allows running Windows applications on Linux systems. However, getting a given Windows application to run on Wine has always been a considerable burden even for highly technical users. To avoid users needing to setup a working 64-bit Wine environment themselves, we created a Docker container with a working environment which includes Skyline and msconvert. Our focus has been on non-interactive, command-line-driven usage because most users who have expressed interest to us in running ProteoWizard tools on Linux were interested in using it in on HPC clusters.

This development makes it practical to directly read vendor data files when running workflows on HPC clusters or on cloud computing resources (both of which are usually Linux-based).

The Docker image is hosted at DockerHub:

https://hub.docker.com/r/proteowizard/pwiz-skyline-i-agree-to-the-vendor-licenses

One of our concerns about running under Wine was whether the vendor file readers would be significantly slower, and we were pleased to find that for most vendor formats this was not the case.

Supported Vendor Formats

Format	Status
ABI T2D	not working
Agilent MHDAC (non-IMS)	working
Agilent MIDAC (IMS)	working
Bruker BAF	working
Bruker FID/YEP	not working
Bruker TSF/TDF	working
Mobilion MBI	working
Sciex WIFF	working
Sciex WIFF2	working
Shimadzu QQQ	working
Shimadzu QTOF	working
Thermo RAW	working
Waters RAW	working
Waters UNIFI	not working

Almost all of ProteoWizard's formats work with Wine. T2D is 32-bit only and requires vendor software to be installed to work, so we do not expect it to work with Wine. The issues with FID/YEP and UNIFI are still being investigated.

Methods

We benchmarked on two nominally equivalent Amazon Web Services instances (m7i-xlarge type), one running ProteoWizard tools natively on Windows, and the other running them in the Docker/Wine container on Amazon Linux. We tested performance in three scenarios:

1. Converting to mzML with msconvert
2. Importing results into a Skyline document
3. Reading spectra with msbenchmark

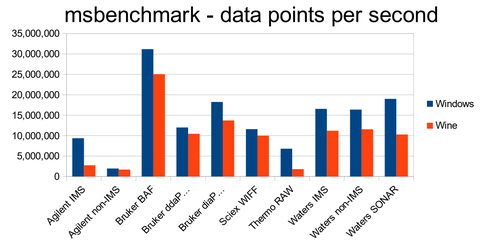
Each scenario stresses different parts of the system's performance. The msconvert scenario tests disk writing performance. The Skyline scenario tests multithreaded processing during import. Finally, msbenchmark does no processing or writing to disk so it is nearly a pure test of vendor reading performance.

Wine vs. Windows Performance

This table shows the relative performance ratios between Wine and Windows for msconvert and Skyline. Red indicates Windows is faster while blue means Wine is faster.

Vendor Format	msconvert	Skyline	msbenchmark
Agilent IMS	1.43	1.17	3.41
Agilent non-IMS	0.99	1.23	1.16
Bruker BAF	1.04	0.83	1.25
Bruker ddaPASEF	1.23	2.27	1.14
Bruker diaPASEF	1.02	1.85	1.33
Sciex WIFF	1.13	1.09	1.16
Thermo RAW	1.66	1.10	3.74
Waters IMS	1.05	*	1.48
Waters non-IMS	1.02	1.30	1.42
Waters SONAR	1.02	1.60	1.85

For most formats, Wine is almost as fast as Windows. There are a few outliers, as well as the occasional stability issue*: on Wine, msconvert was able to convert the Waters IMS data, but Skyline repeatedly crashed when trying to import it. Thermo RAW and Agilent IMS data are outliers that are the most affected by Wine's performance differences. They are most pronounced in the msbenchmark ratios, indicating the differences are starkest for the vendor API calls rather than disk writing in msconvert or multithreading in Skyline. On the other hand, importing Bruker PASEF data into Skyline is clearly less performant on Wine than on Windows, but this is apparently not due to vendor API performance.



We can also look at the performance of each vendor in absolute terms: number of m/z and intensity data points read per second (a more accurate metric than number of spectra, since spectra vary considerably in density between vendors). Agilent non-IMS performance suffers because its vendor API returns profile data with tens of thousands of superfluous zero samples, which ProteoWizard must filter out.

Conclusions

The ProteoWizard Docker/Wine image is already a successful resource for users wishing to convert their MS data on Linux systems. It has been "pulled" over a million times from DockerHub*. To our knowledge, there is no other MS analysis or conversion software capable of reading as many vendor formats on Linux.

It is worth noting that our Docker image is only available for AMD64 instruction set processors (due to depending on vendor DLLs only built for AMD64). Some users have requested support for running on ARM[64] processors though and we would like to work with community members with such hardware to allow our Wine solution to work via AMD64 emulation.

Docker's centralized public repository allows easy access to our image from anywhere in the world. We are aware that many shared HPC systems do not allow users to run Docker containers, but many users have reported to us that our Docker image is compatible with the Singularity container system which usually is allowed on HPC systems.

We are grateful to Steffen Neumann (IPB Halle), Rene Meier (IPB Halle), Pablo Moreno (EMBL-EBI), and Pierrick Roger (CAE) for their work on their own Docker/Wine ProteoWizard container which convinced us that the combination of Wine and Docker would make it simple and easy to run ProteoWizard tools on Linux. We are also grateful to the tremendous work done by the Wine and Docker teams: we stand on the shoulders of giants.

*Counting pulls from the original chambm location for the container; it has since been moved to the "proteowizard" organization.

Conflict of Interest Statement

The MacCoss Lab at the University of Washington has a sponsored research agreement with Thermo Fisher Scientific. Michael J. MacCoss is a paid consultant for Thermo Fisher Scientific. Five major mass spectrometry vendors (Agilent, Bruker, Shimadzu, Thermo Fisher Scientific and Waters) provide financial support for the development and maintenance of Skyline and ProteoWizard.

References

[1]Kessner D, Chambers M, Burke R, Agus D, Mallick P. ProteoWizard: open source software for rapid proteomics tools development. Bioinformatics. 2008 Nov 1;24(21):2534-6. doi: 10.1093/bioinformatics/btn323. Epub 2008 Jul 7. PMID: 18606607; PMCID: PMC2732273.
[2]MacLean B, Tomazela DM, Shulman N, Chambers M, Finney GL, Frewen B, Kern R, Tabb DL, Liebler DC, MacCoss MJ. Skyline: An Open Source Document Editor for Creating and Analyzing Targeted Proteomics Experiments. Bioinformatics. 2010;26(7):966–8. doi: 10.1093/bioinformatics/btq054