Alchemy: A Web 2.0 Real-time Quality Assurance Platform for Human Immunodeficiency Virus, Hepatitis C Virus, and BK Virus Quantitation Assays

Emmanuel Agosto-Arroyo, Gina M. Coshatt, Thomas S. Winokur, Shuko Harada, Seung L. Park

1Department of Pathology, University of Alabama at Birmingham, Birmingham AL, USA

Received: 15 September 2016    Accepted: 01 March 2017    Published: 10 April 2017

Abstract

Background: The molecular diagnostics laboratory faces the challenge of improving test turnaround time (TAT). Low and consistent TATs are of great clinical and regulatory importance, especially for molecular virology tests. Laboratory information systems (LISs) contain all the data elements necessary to do accurate quality assurance (QA) reporting of TAT and other measures, but these reports are in most cases still performed manually: a time-consuming and error-prone task. The aim of this study was to develop a web-based real-time QA platform that would automate QA reporting in the molecular diagnostics laboratory at our institution, and minimize the time expended in preparing these reports.

Methods: Using a standard Linux, Nginx, MariaDB, PHP stack virtual machine running atop a Dell Precision 5810, we designed and built a web-based QA platform, code-named Alchemy. Data files pulled periodically from the LIS in comma-separated value format were used to autogenerate QA reports for the human immunodeficiency virus (HIV) quantitation, hepatitis C virus (HCV) quantitation, and BK virus (BKV) quantitation. Alchemy allowed the user to select a specific timeframe to be analyzed and calculated key QA statistics in real-time, including the average TAT in days, tests falling outside the expected TAT ranges, and test result ranges.

Results: Before implementing Alchemy, reporting QA for the HIV, HCV, and BKV quantitation assays took 45–60 min of personnel time per test every month. With Alchemy, that time has decreased to 15 min total per month. Alchemy allowed the user to select specific periods of time and analyzed the TAT data in-depth without the need of extensive manual calculations.

Conclusions: Alchemy has significantly decreased the time and the human error associated with QA report generation in our molecular diagnostics laboratory. Other tests will be added to this web-based platform in future updates. This effort shows the utility of informatician-supervised resident/fellow programming projects as learning opportunities and workflow improvements in the molecular laboratory.

Keywords: BK virus assay, hepatitis C virus assay, human immunodeficiency virus assay, molecular quality assurance, quality assurance, quantitation assays

Introduction

We are in a new world of molecular diagnostics and targeted therapies. Nucleic acid amplification technologies have allowed for breakthroughs in both detection and load measurement of pathogenic viruses. Wide-scale automation of these technologies – most notably quantitative real-time polymerase chain reaction – has increased the precision and throughput of clinical testing to the point that molecular diagnostics methods have become the gold standard for monitoring viral loads of multiple clinically significant infectious agents. Quantitation of human immunodeficiency virus (HIV), hepatitis C virus (HCV), and BK virus (BKV) are among the most significant of these assays. As demand for these molecular tests has increased, so too has the amount of data generated by the molecular diagnostics laboratory. Truly genomics-aware electronic medical records (EMR) and laboratory information systems (LISs) do not yet exist. The sheer amount of data generated by
even a single molecular diagnostics test provides a unique challenge for proper quality analytics. One of the most important quality analytics measurements in the clinical laboratory is turnaround time (TAT), for both clinical and regulatory reasons.\textsuperscript{[7,8]} Unfortunately, in many clinical laboratories, where automated TAT calculating modules are not available, the process of quality assurance (QA) reporting on TAT usually involves manual steps (e.g., hand cleanup and further manipulation of data in a program like Microsoft Excel), resulting in a time-consuming and often error-prone process that consumes the time of highly trained laboratory staff.

LIS and EMR technologies – which have the stated purposes of automation, centralized data warehousing, and write-once-deploy-everywhere reporting – have paradoxically caused an explosion of manual processes, most of them are workarounds to both perceived and actual inflexibility in deployed systems. Denial of appropriate database access, centralized bureaucratic control of analytics assets, and personnel shortages secondary to tightened budgets all play a part in creating a situation where QA reporting that could be automated and performed in real-time within the boundaries of the LIS is often impossible.

Given the technical and administrative limitations that were present at our institution, we chose to develop a web-based real-time QA platform that would automate QA reporting in the molecular diagnostics laboratory. We chose to do so for several reasons: (1) reports pulled relevant data from the LIS did not aggregate or perform meaningful analytics on said data; (2) skilled laboratory personnel were wasting up to 45 min/test/month on hand manipulation of data sets to generate appropriate QA reports; (3) the existence of pathology informatics expertise on the molecular diagnostics faculty; (4) the suspicion that the existing manual process was susceptible to human error; and (5) lack of financial and technical resources would make addressing our issues a low-priority item on the institutional roadmap.

**Methods**

VMware ESXi 4.1.0 (Virtual Machine Hypervisor Software, VMware Incorporated, Palo Alto, CA) was installed on a Dell Precision 5810 (Intel Xeon E5-1620 v3; 16GB DDR3 ECC SDRAM; 256GB LiteOn SSD; 1TB Western Digital HDD) server. This server was isolated within the internal hospital intranet, with no physical or network accessibility from the university intranet or internet at large. This allowed for a server that could spin up individual experimental virtual machines at will, yet did not pose a security risk. To safeguard against a physical security breach, the server was located within a locked room and password protected, with two-factor authentication through RSA SecurID. The SSD and HDD were hardware encrypted.

The molecular diagnostics fellow was given a 1-month long elective informatics rotation. This rotation was an extended version of a 2-week long flipped classroom informatics boot camp that we had previously run for students of our institution’s Medical Scientists Training Program.\textsuperscript{[9,11]} During this rotation, the fellow received specialized instruction in the following topics:\textsuperscript{[12]}

- Virtual machine theory and practice
- Stack-based computing
- UNIX-like operating systems and POSIX standards
- Web servers, HTTP, HTTPS, proxies, and reverse proxies
- Relational database management systems, SQL, database normalization, and denormalization
- Server-side (PHP) and client-side (Javascript) scripting
- User interfaces and human-computer interaction paradigms
- Data file parsing, error correction, and fault tolerance.

On initial investigation, the fellow ascertained that QA reports were being manually pulled from the LIS (Cerner Millennium) through a custom script written in Cerner Data Explorer by a member of our institution’s IT staff. These reports were:

- (1) extremely verbose;
- (2) did not directly contain the reportable data elements;
- (3) could not be run at arbitrary time periods, and
- (4) were not regular in columnar structure, with multiple irregularities detected within each report run. Laboratory personnel was forced to dump all rows (usually numbering into the thousands) into an Excel Spreadsheet, regularize the data by hand, and then apply different mathematical transformations on the sanitized data to arrive at the finalized QA reports. This took an average of 45–65 min per test per month – a small amount of time individually, but an untenable amount of time when the number of test types performed by our molecular diagnostics laboratory was considered.

The fellow, mentored by a member of the pathology informatics faculty, performed a detailed analysis of the source data files. The following irregularities were found:

- (1) the presence of periodic “test” results that were generated by the upstream EMR on patients that existed in the system solely for the purposes of engineering QA and were therefore not real;
- (2) certain (but not all) test results that fell within a critical high/critical low range took up two rows, with the second row consisting solely of an all caps comment block
- (3) after such a row, there was a $+2$ columnar “frameshift” of the next row;
- (4) many rows were extraneous, as they were data points on tests either not truly offered by our molecular diagnostics laboratory or (more rarely) on tests that had been placed in error.

All of the above, perhaps save the first, could have been addressed with a cleanup of the Cerner Data Explorer custom script. Our team initially asked our institutional IT for this, but due to the lack of resources and existence of a manual workaround, this request was given low priority. However, even had all of these issues been addressed, it would have been impossible to create a single script that performed the necessary
calculations and transformations that we required to arrive at a finalized QA report. Cerner Custom Language (CCL) is a database query language that is best understood as a reskinned superset of SQL, with all of the limitations of SQL. Specifically, conditional branching is impossible in CCL without the implementation of recursive common table expressions; this is not generally done for multiple reasons, not the least of which is performance. As such, we were left with a source data file from which we could derive all necessary elements, but which required a large amount of error correction to be machine usable.

We chose to implement a Linux, Nginx, MariaDB, PHP (LEMP) virtual machine, utilizing the following components:
- Ubuntu Linux 12.04 LTS
- Nginx 1.5.6
- MariaDB 5.5.34
- PHP 5.4
- jQuery 1.11
- Twitter Bootstrap 3.

Once the virtual machine was fully configured, we proceeded to design a Web 2.0 real-time QA platform. At periodic intervals, source data autogenerated as a comma-separated value (CSV) file from Cerner Data Explorer would be pulled into the virtual machine utilizing HTTP POST. Utilizing a 1-year long historical backlog of such CSV files, we designed and implemented a robust error and duplication removal program in PHP to run as a first pass. The result of this PHP program was a clean data file that only contained rows with known good data, in a regularized format.

This sanitized data was then passed to a PHP-SQL script. This script inserted the rows directly into a permanent SQL database, utilizing SQL prepared statement constructs to lower computational and I/O overhead by an order of magnitude as compared to the standard declarative method, while also providing a final duplication check. Lines with errors were dumped into a temporary error table for later analysis.

Once the rows were housed in the database, this database could be queried at will, for arbitrary tests and arbitrary time spans. We wrote a complete user interface for both file upload/parse and querying/reporting utilizing PHP, SQL, HTML, CSS, jQuery, and Bootstrap [Figures 1 and 2]. Utilizing XMLHttpRequest and other asynchronous JavaScript as XML programming techniques, we constructed a reporting framework in which arbitrary timeframes could be defined, analyzed, and output as fully formatted College of American Pathologists compliant QA reports in real-time. This real-time framework, which we code-named “Alchemy,” takes in as user inputs the test type and timeframe period [Figure 3]. Once these are defined, Alchemy generates a summary report which calculates key QA statistics in real-time, including the average TAT in days, the number of tests falling outside the expected TAT ranges and their corresponding percentages [Figure 4]. In addition, a results ranges table is generated, summarizing the number of tests for each period of time that falls into specific predefined ranges, as defined by the medical director of the molecular diagnostics laboratory [Figure 5]. A raw data table is
also available at the end of each report; it includes the specific accession number, specific ordering and reporting dates, TAT, and actual test result.

**RESULTS**

A summary of the required activities to obtain the data and prepare the QA reports along with the amount of time invested in each activity is shown in Table 1. Before implementing the Alchemy platform, reporting QA for the HIV, HCV quantitation, and BKV quantitation assays took 45–65 min of personnel time per test every month. As expected, report inquiry and the summary of data were the most time-consuming tasks. The report inquiry was completely dependent on the LIS. The summary of the data required the user to separate the pertinent information for each specific test from a file including all tests performed during the period of interest and multiple result values. In addition, QA parameters calculations were manually performed.

With Alchemy, the time spent in preparing QA reports decreased to 15 min total (regardless of a number of tests) per month. Discrepancies in between the previous manual “gold standard” and Alchemy were found only three times in the first full quarter of operation, and all three times, the discrepancy was found to be due to either human error or data duplication in the previous gold standard. Perhaps, even more impressive was the fact that during the first full month of operation, there was a large discrepancy between the number of tests reported by Alchemy and the number of tests counted by the official report from the LIS. A bug was identified in the way that the LIS report was calculating data elements, and Alchemy’s numbers were proven to be correct by manual reconciliation.

**DISCUSSION**

Viral disease testing is intended to identify and accurately diagnose the infection and establish the therapeutic and follow-up approach. The clinical utility of nucleic acid technologies for viral quantitative tests is well established. These molecular technologies allow monitoring changes in viral load during treatment and predicting the response to therapy. In the era of “test and treat,” it is of utmost importance to provide the test results in a timely manner.

QA programs are developed to assess the efficacy of the laboratory in providing the test result within the expected period of time. The use of quality control tools in the molecular microbiology assays is crucial to ensure the accuracy of results, appropriate patient management, and error detection. The auditing of laboratory efficacy provides a way for the laboratory personnel to know if specimens are being handled adequately and may provide awareness into possible areas of deficiency, ideally allowing for rapid process improvement. Unfortunately, as demonstrated by our experience, when the preparation of such QA reports is manual, the time expenditure becomes cost-prohibitive, and errors arise.

LISs are critical for high-quality healthcare service provision and the need for these systems is growing. With the advent of better computer systems, an increasing number of healthcare institutions rely on computer programs to handle the burdensome processing of data and to provide timely feedback in a manner that is easily understood and readily interpreted by the laboratory staff. Laboratory process automation has greatly improved test throughput, while driving down the TAT, enabling critical results to reach physicians rapidly for improved clinical outcomes.
LIS’s are implemented on the premise that they meet these technological and workload demands. Ideally, they enable laboratory staff to monitor the analytical and quality processes while minimizing errors. Automatic calculation of TAT and other laboratory statistics are currently offered by several vendors, either integrated into the core LIS software or as modules that can be customized to accommodate the laboratory needs. Unfortunately, different barriers may limit the availability of these options to laboratories, including economical as well as infrastructural. Even though it has been decades since the implementation of the first LIS, the dream of real-time data analysis through the data warehoused in the LIS has largely remained out of reach.

LEMP stacks have been used for the development and deployment of many web applications. Its modular nature offers a flexible “applicability.” LEMP stacks use an open source Unix-like operating system as the base for the stack components and Nginx as a reverse proxy server for HTTP, HTTPS, SMTP, POP3, and IMAP protocols. Nginx exhibits high currency, high performance, and low memory usage as compared with its peers, making it ideal for a project like Alchemy. PHP is a widely used scripting language which can run on various platforms and is compatible with almost all currently used servers. PHP supports a wide range of databases and can generate dynamic page content, create and manipulate files on the server, modify the database, control user access, and even encrypt data. PHP is easy to learn; it has a basic syntax and is excellent for beginners. Importantly, the contents of the LEMP stack are free and easily accessible, which provides an incentive for those interested in creating an application customized for the individual needs and in an affordable fashion.

We have previously reported the use of web tools not only in genomics and research but also as an excellent tool for pathology resident’s education. We followed a similar approach in this study. In creating Alchemy, we not only created a powerful real-time tool for quality improvement in our molecular diagnostics laboratory but also provided a trainee with the rare opportunity to drive clinical improvement through direct design, implementation, and deployment of a clinically viable informatics-driven system. This approach per se is not utilizing any novel methodology. However, we rather intended to show a feasibility of process how to utilize simple informatics approach by a resident/fellow supervised by an informatician.

By using Alchemy, a significant decrease in the time and the human error associated with QA report generation in our molecular diagnostics laboratory has been observed. Alchemy has proved to be an effective easy to use tool as well as an inexpensive, approachable, and effective teaching tool for informatics. We have used Alchemy for the calculation of TAT for infectious disease tests in the molecular laboratory. However, its applications are not limited to these tests. With minor programming modifications to the code, more tests can be added to the menu. We have also been able to obtain additional information from the original data with the use of Alchemy, including the number of tests with different result ranges in a period of time which helps for statistical and monitoring purposes. We plan to add more tests to the platform’s menu and include tests from other laboratory divisions in future updates. This effort shows the utility of informatician-supervised resident/fellow programming projects as learning opportunities and workflow improvements in the molecular laboratory – a utility that we have managed to demonstrate many times over. Alchemy is also a feasible option for organizations facing similar issues as ours and looking for a cost-effective way to improve the completion of QA reports. We hope to continue utilizing these flipped classroom, closely mentored training methodologies to drive continued success in pathology informatics at our institutions.

CONCLUSIONS

Herein, we developed a web-based real-time QA platform that automated QA reporting in the molecular laboratory. Alchemy is a feasible option for organizations facing similar issues as ours and looking for a cost-effective way to improve the completion of QA reports. We hope to continue utilizing these flipped classroom, closely mentored training methodologies to drive continued success in pathology informatics at our institutions.

Financial support and sponsorship
Nil.

Conflicts of interest
There are no conflicts of interest.

REFERENCES

1. Costello M, Sabatini LM, Yungbluth M. Viral infections. In: McPherson RA, Pincus MR. Henry’s Clinical Diagnosis and Management by Laboratory Methods. 22nd ed. Philadelphia, PA: Elsevier; 2011. p. 1037–64.
2. Branson BM, Owen SM, Wescoslowksi LG, Bennett B, Werner BG, Wroblewski KE, et al. Laboratory Testing for the Diagnosis of HIV Infection: Updated Recommendations. Atlanta, GA: Center for Disease Control and Prevention; 2014.
3. Centers for Disease Control and Prevention (CDC). Vital signs: HIV prevention through care and treatment – United States. MMWR Morb Mortal Wkly Rep 2011;60:1618–23.
4. Butcher A, Aslam S, Hemyard P, Cowen U, Heilek G. HCV RNA detection in HCV antibody-positive patients with the COBAS AmpliPrep/COBAS TaqMan HCV test, v2.0 in comparison with FDA-approved nucleic acid tests. J Clin Virol 2014;60:336–40.
5. Centers for Disease Control and Prevention (CDC). Testing for HCV infection: An update of guidance for clinicians and laboratorians. MMWR Morb Mortal Wkly Rep 2013;62:362–5.
6. Nankivel BJ, Renthawa J, Jeffreys N, Kable K, O’Connell PJ, Chapman JR, et al. Clinical utility of urinary cytology to detect BK viral nephropathy. Transplantation 2015;99:1715–22.
7. Ferreira-Gonzalez A, Wilkinson DS, Bluth MH. Establishing a molecular diagnostics laboratory. In: McPherson RA, Pincus MR, editors. Henry’s Clinical Diagnosis and Management by Laboratory Methods. 22nd ed. Philadelphia, PA: Elsevier; 2011. p. 1314–29.
8. College of American Pathologists. Northfield, IL: CAP Accreditation
Program-Microbiology Checklist; 2011. Available from: http://www.cap.org. [Last accessed on cited on 2014 May 07].

9. Kennell T, Laufer V, Lorenz R, Park S. This Is Your Brain on Informatics: A Total-Immersion Data Sciences Course for the Next Generation of Informaticists (Platform Presentation). Pathology Informatics Summit. Pittsburgh, PA, USA; 2014.

10. Park S, Kennell T, Lorenz R. One Year of Fried Brains: Informatics as a Driver of Cultural Change at an NIH Medical Scientists Training Program (MSTP). Pathology Informatics Summit. Pittsburgh, PA, USA; 2015.

11. Park S, Parwani A, Macpherson T, Pantanowitz L. Use of a wiki as an interactive teaching tool in pathology residency education: Experience with a genomics, research, and informatics in pathology course. J Pathol Inform 2012;3:32.

12. Park S, Balis U, Pantanowitz L. Computer fundamentals. In: Pantanowitz L, Tuthill JM, Balis U, editors. Pathology Informatics: Theory and Practice. Chicago: ASCP Press; 2012.

13. Cerner Corporation. Discern Explorer Main Help. uCern; 5 October, 2015. Available from: https://www.wiki.ucern.com/display/1101discernHP/Discern+Explorer+Main+Help. [Last cited on 2015 Oct 22].

14. Orta Mira N, Guna Serrano MR, Gimeno Cardona C, Pérez JL. Quality control in molecular microbiology. Enferm Infecce Microbiol Clin 2008;26 Suppl 9:2-7.

15. Harrison JP, McDowell GM. The role of laboratory information systems in healthcare quality improvement. Int J Health Care Qual Assur 2008;21:679-91.

16. Tan IK, Jacob E, Lim SH. Use of computers in quality assurance of laboratory testing. Ann Acad Med Singapore 1990;19:724-30.

17. LEMP Stack Info. LEMP Stack Info.; 2015. Available from: https://www.lemp.io/. [Last cited on 2015 May 24].

18. W3Schools. Norway: Refsnes Data, Inc.; 1999-2015. Available from: http://www.w3schools.com/php/default.asp. [Last cited on 2014 May 07].

19. Hutchinson L. Web Served: Get Your Database On. Ars Technica; December, 2012. Available from: http://www.arstechnica.com/information-technology/2012/12/web-served-part-4-get-your-database-on/. [Last cited on 2014 May 07].

20. Cain M, Drwiega J, Ping Z, Park S. Constellation: A Web 2.0 Adult Autopsy Reporting and Analytics Platform. Pathology Informatics Summit. Pittsburgh, PA, USA; 2015.

21. Soni A, Harada S, Park S. HCV Genie: A Web 2.0 Interpretation and Analytics Platform for the Versant Hepatitis C Virus (HCV) Genotype Line Probe Assay Version 2.0. Pathology Informatics Summit. Pittsburgh, PA, USA; 2015.

22. Siniard RC, Harada S, Park S. Asterism: A Web 2.0 Molecular Diagnostics Templating, Reporting, and Analytics System. Pathology Informatics Summit. Pittsburgh, PA, USA; 2015.