

SYSTEM INTEGRATION EFFORT ON MagViz A LIQUID EXPLOSIVE DETECTION DEVICE

Martin Pieck*, Jeff O. Hill, John F. Power, Los Alamos National Laboratory, NM 87545, USA

Abstract

Los Alamos National Laboratory (LANL) technologists have adapted Magnetic Resonance Imaging (MRI) technology from the familiar medical device to create MagViz a new tool that distinguishes potential-threat liquids from the harmless shampoos and sodas a regular traveler might take aboard an aircraft. Funded by the U.S. Department of Homeland Security (DHS), the MagViz system is a new, ultra-low-field MRI approach first designed for brain imaging, but with a unique variation. Linked with a computer database, MagViz can now reliably identify some 50 liquids from their chemical fingerprints.

The underlying technology for MagViz has been developed by a LANL Physics Team for an experimental laboratory environment. The instrumentation and controls team took the collection of individually developed, highly incoherent, multiple platform hardware and software applications (data analysis, calibration programs, data acquisition system, squid control system [1], and the waveform generation system), developed additional applications (conveyer belt, camera system, graphical user interface, and power supply system), and integrated them into one ready for production use. This paper will describe the integration effort, the challenges faced and the lessons learned when these systems were integrated.

BACKGROUND

A terrorist plot to blow up aircraft with liquid explosive components carried aboard was thwarted in London in 2006, prompting today's three-ounce restriction for carry-on liquids. MagViz is a direct response to that type of threat. Using MRI, MagViz identifies chemicals by measuring the magnetic interactions of their protons with the local molecular environment, despite these fields being one thousand to one million times lower than those of MRI machines used in hospitals. MagViz then matches the proton signals from scanned items to those in a database of benign and threat liquids. The technology not only detects liquid explosives in less than 60 seconds, it scans multiple containers simultaneously, detects volumes as small as one millimeter and sees through certain types of metal containers. [2] This technology received a 2009 R&D 100 Award.

INTRODUCTION

In August of 2007, an interdisciplinary team from science and engineering divisions across LANL was assembled including teams from Applied Modern Physics, Mechanical Design Engineering, Accelerator and Beam Science, Radio Frequency Engineering, Dynamic and

* pieck@lanl.gov

Energetic Materials, Space Instrumentation Systems, Applied Engineering Technology, Safeguard Science and Technology, Instrumentation and Controls, and the National High Magnetic Field Laboratory. In 13 months this team took the MagViz concept to a production level product. The system consisted of components that were independently developed and implemented by different members of the original project team using different techniques, programming languages, and operator interfaces. That system had served well in the test lab environment for the purpose of "proof-of-principle" but, as seen in the figure below was for the expert users only.

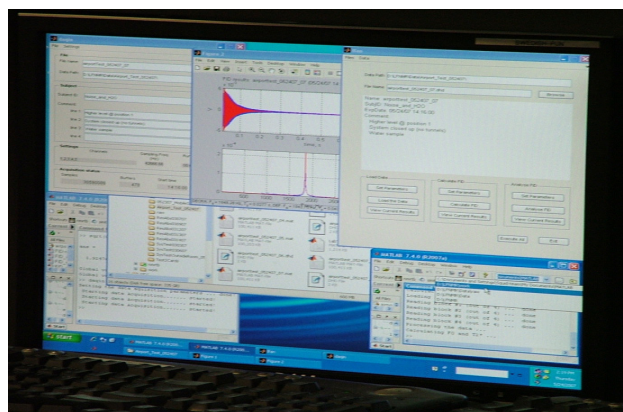


Figure 1: Test system user interface screens.

Setup and sequencing of operation required detailed knowledge and many user interactions with the individual program parts. For a production system that would be operated by the Transportation Security Administration (TSA) employees this system needed improvement considering the goal of a 24/7 operation where reliability, ease of operation, and maintainability have the highest priority.

The starting point of the integration effort was a suit of existing components to be integrated:

- Squid Control (Software (SW): Proprietary Protocol, Hardware (HW): RS 232 Hardware Interface)
- Data Analysis (SW: MATLAB, HW: None)
- Data Acquisition (SW: C-Program, HW: Custom)
- Waveform Generation (SW: MATLAB, C++ Driver, National Instrument (NI) 6733 Analog Output Card)

It should be noted that there was originally no direct software communication between the different components above. In an effort to enhance the system the following additional functionalities needed to be integrated; replace the current system of batteries with Power Supplies, integrate a Conveyer Belt via a programmable logic controller (PLC), integrate a Visual Camera System to compare sample tray with analysis results, and provide an Operator Interface for easy user interaction.

HARDWARE SYSTEM OVERVIEW

Integrating all the above mentioned hardware components resulted in the network architecture design as shown in Figure 2 and the physical location of the equipment as shown in Figure 3.

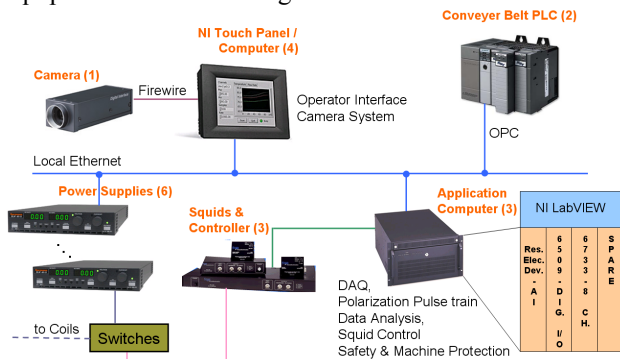


Figure 2: Hardware architecture of MagViz control and data managing system.

The communication backbone for the hardware architecture is a local Ethernet network. Attached to the network are three computers. First, a custom built application computer (Fig. 2 & 3 No. 3) with two AMD Opteron DP processors, Model 2350 (8 processors total) and four disks for a RAID 0+1 configuration running Windows XP. Second, a NI Touch Panel/Computer PPC-2015 15in. Panel, Windows XP, 2.0 GHz Pentium 4, 512 MB and 40 GB HD (No. 4). Attached to the Touch Panel via Fire-Wire is a camera (No. 1) scA1000-30fc color CCD, 1034 x 779, 1/3" format, 30fps with a DF6HA-1B 6mm, 1/2" format, f1.2, 100mm MOD, filter M27 P=0.50. Finally, a PLC (No. 2) for the conveyor belt control is connected to the Ethernet as well. The Power Supplies (No. 6) for the magnetization coils are controlled through the application computer. Fig. 3 No. 5 shows the magnetization system and part of the data acquisition system.

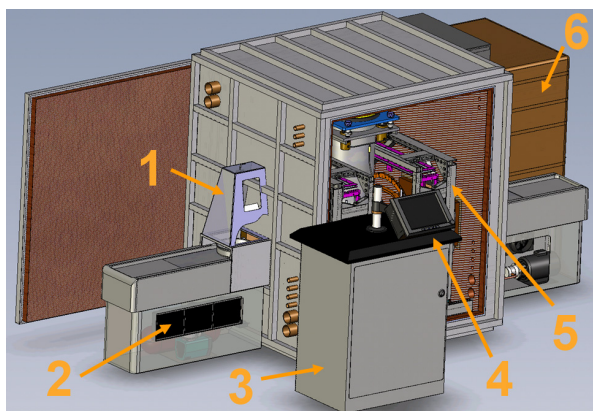


Figure 3: Hardware architecture of MagViz control and data managing system.

SOFTWARE SYSTEM OVERVIEW

Naturally the hardware architecture goes hand in hand with the software architecture. After the consideration of all aspects of the project (scope, cost, and schedule) the

easiest integration development environment appeared to be National Instrument's LabVIEW. It provides an integrated development environment for different hardware and software solutions and easy to develop operator interfaces. Additional reasons that led to the decision to use LabVIEW are listed below:

- LabVIEW is commonly used for Data Acquisition, Instrument Control, Test Automation, Analysis and Signal Processing and is well recognized in industry.
- LabVIEW is multithreaded and runs on 64-bit Dual and Quad-Core Intel® Processors which are faster than the computers previously being used.
- The integration of the camera and availability of supporting toolboxes is a tremendous time saver.
- LabVIEW MathScript allows running native MATLAB programs (like the one currently used for the data analysis) in the LabVIEW environment.
- All relevant hardware proposed was available with LabVIEW Drivers, the development of which is usually a bottle neck in software development.
- Integration for both production & experimental use requires an agile system like LabVIEW.

Figure 4 gives an overview on how the LabVIEW development environment (light blue rectangle) provides integration support (different colored boxes) of the different MagViz components (ovals):

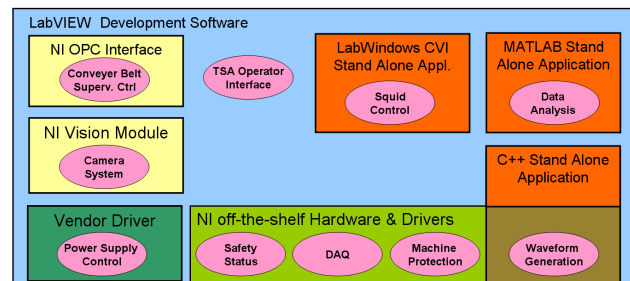


Figure 4: Development software environment.

SYSTEM INTEGRATION

While the above Figure suggests that the integrations must have been straightforward the project encountered numerous challenges:

- LabVIEW does allow running native MATLAB code under their MathScript interface; however, not all functionality that is available under MATLAB is supported by LabVIEW's MathScript. NI offered to implement the missing functionality on a short notice but the project opted not to take this route since future developments in MATLAB could imply NI dependency.
- MATLAB programs needed to communicate with LabVIEW. The native MATLAB interface had for our needs a typedef mismatch which required a custom C-language module to establish communication between the programs from the different vendors.

- LabVIEW provides shared variables for easy communication between LabVIEW programs on the same or even different computers. However, tying shared variables directly to buttons on the user interface allowed an intermittent race condition to occur. The probable cause was confusion over where the variable was hosted versus the communication path for the variables' updates. Looking back, a single powerful application computer and a user interface exported to the touch screen would have saved a lot of time.
- Source Code Control had to deal with different types of source codes (graphical and text based). Our version (8.5) of the stand-alone LabVIEW differencing utility crashed and this made it difficult to review commits. Each version of LabVIEW has a unique binary source code protocol creating additional challenges for source code control.
- LabVIEW did run on 64-bit architecture but not all hardware and in particular our DAQ board did not have 64-bit drivers available at that time.
- Difficult but essential for the project was the error handling in such a heterogeneous system. Low-level error messages were passed through the different software layers up to the user interface. This helped in understanding and eliminating problems.

MagViz IN ACTION

The MagViz Threat Detection Software State Diagram is shown in Figure 5.

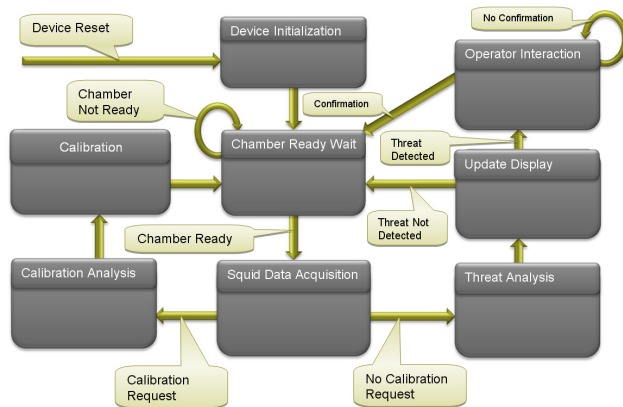


Figure 5: Threat detection software state diagram.

In short, at startup the machine goes through an initialization phase and the chamber is tested for its readiness to accept a sample tray. If it is ready the tray is transported via the conveyer into the chamber. Before it enters the chamber the camera takes a picture of the tray (Figure 6). After the tray reaches its destination in the center of the enclosure several magnetization data acquisition activities take place. If MagViz finds a chemical designated as a threat, the machine will mark the container with a red dot on the screen (Figure 7). Harmless substances get a green dot, and if the machine can't identify the liquid, a yellow dot appears, indicating that further inspection is needed. As new threats emerge,

the database can be augmented. In MagViz's current incarnation, the entire process takes a minute. Although the MagViz system is capable of producing complex three-dimensional images, TSA agents would see the non-ambiguous color-coded readout that quickly gives them the information they need. A prototype MagViz machine was unveiled at the Albuquerque International Sunport in December 2008, where it underwent field testing. Should the development process continue successfully, machines could be in airports by 2012.



Figure 6: Image of sample tray.

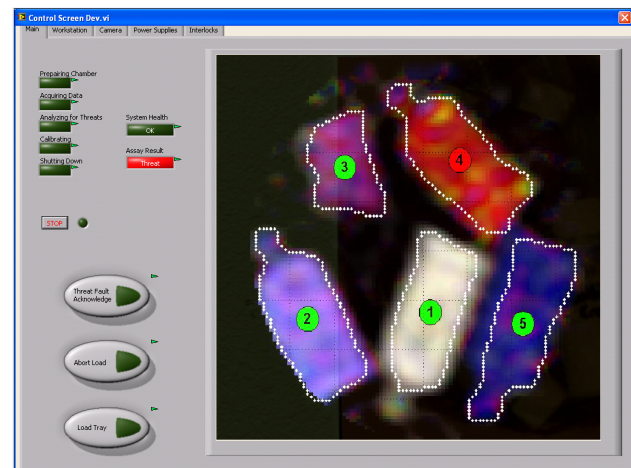


Figure 7: Prototype system user interface screens.

CONCLUSION

Integration of heterogeneous software and hardware is difficult to accomplish efficiently. Effective integration occurs only with significant upfront investments in planning of a project like MagViz as a whole. This type of integration not only increased efficiency, but also helped to improve the outcome.

REFERENCES

- [1] STAR Cryoelectronics is a leading manufacturer of SQUID sensors and advanced PC-based SQUID control electronics. <http://www.starcryo.com/>.
- [2] Owens, Mike, "Expediting carry-on screen for air travel." Los Alamos National Laboratory Currents, Los Alamos, New Mexico (June 2009).