
TRIGYN
technologies

1. Overview

Like many applications today, the Human Machine Interface (HMI) landscape has undergone tremendous change. The role of the HMI and the functions it performs is constantly being evaluated. The question being asked is whether or not HMIs can survive in today's industrial environment. From a historical perspective HMIs have survived by adopting evolving technologies to satisfy customer needs. A look back over the past 15 years not only demonstrates this adoption but highlights the increasingly important role that HMIs have played. HMIs have grown beyond simple graphical representation of the factory floor to today's HMIs that bridge the gap between automation and IT systems. A look into the future promises that HMIs will continue to adopt technologies that will play an even greater role in e-Manufacturing.

Human Machine Interface systems have come a long way from being simple push buttons, thumbwheers and selector switches to robust client-server applications enabling e-Manufacturing.

HMI Systems: A look Back

The earliest forms of „soft" Human Machine Interface systems were text based, followed by programmable DOS-based graphic terminals.

With the evolution of Windows though, HMI systems got enhanced to sophisticated GUI interactive applications.

2. Human Machine Interface: A Look Back

At the onset, operator interfaces were merely push buttons, thumbwheels and selector switches that eased the operation of automated manufacturing. This was followed by text based and programmable DOS-based graphic terminals. These programmable operator interfaces allowed production lines quicker changeovers for producing derivatives of the standard product, enabling manufacturers' to offer its customers more choices.

This was followed in the late 80s with simple Man Machine Interface(MMI) applications which were introduced on Windows based PCs. These applications provided simple interactions with industrial processes. As these enhanced graphic applications gained market acceptance, they were gradually recognized from MMI to Human Machine Interface(HMI).

With the development of HMI systems along with an evolution in GUI technologies, functions like process alarming, trending and operator security became a basis to provide better local control and improve machine performance. The evolution of the underlying operating systems for HMI continued with Windows moving to 32-bit architectures. This led to HMIs offering advanced features like

- Alarming functions for sorting based on priority, corrective action to provide better operation data
- Trending representation of both historical and real-time data values. This resulted in quick operator response times to adjust process parameters.
- Password protection schemes provided multiple access levels for operators, process engineers and product managers allowing Each individual could perform their job specific functions while limiting their ability to inadvertently upset the process

To summarize, HMIs were instrumental in increasing machine utilization.

During these early stages of HMIs, the data associated with alarms and trends as

well as additional functionality such as recipe management systems were stored in proprietary formats. Only the tools provided by the HMI could access this data. This led to manufacturers incurring additional costs in maintaining and navigating through proprietary software that most HMIs were designed around. This needed more open architectures to derive useful information out of this data.

3. Human Machine Interface: The Present

To keep up with the evolving with market demands, HMI systems needed to undergo a paradigm shift from merely integrating human operators from the machines they operate, to integrating actual manufacturing processes themselves. To be an integration platform, HMIs need to integrate with the plant floor control systems from multiple vendors including PC based control systems from a variety of vendors. Integrating plant floor control systems is a key area where HMIs have been used. HMI vendors who offer PLC control systems, tightly integrate individual system components (HMI and Ladder Logic, etc.), thus improving configuration efficiencies while reducing development errors.

The Present:
How **M**anufacturing **I**ntegrates
The current demands of plant floor environments have helped HMI systems develop from being monolithic and stovepipe applications, to high end client server systems built on flexible and standard frameworks.

Integration into heterogeneous plant floor control systems through standards such as OPC (OLE for Process Control) has minimized the connectivity uncertainties of mixing HMIs with third party controllers. HMIs have adopted Microsoft standards for data movement and representation. HMIs today are built on open databases, many offering access to the data through ODBC, or Open DataBase Connectivity. HMIs can easily share data with office applications such as MS Excel for reporting, MS Word for embedded help documents and MS Access for Recipe Systems.

Given the new applications for HMIs, and HMIs' growing importance, these systems are expected to be running 24/7 with zero to minimum latency. This has introduced a need for client-server architectures wherein HMIs must act as servers and their backend databases must support redundancy. HMIs must support automatic switching of clients, redundant communications to the plant control systems and in some applications redundant controllers.

Current HMI systems need to get information off the factory floor and into the hands of decisions makers, thus making information exchange a core expectation. The use of standard networking technologies and topologies in IT, in conjunction with the open HMI systems in automation, are a natural fit. HMIs now provide web-based technologies, ones that incorporate the local operator security systems as well as the remote Internet firewall security. By using web-based technologies, real-time process representation enables manufacturers to provide remote production troubleshooting by its process experts. This ensures quicker and accurate resolution to system disturbances.

4. Human Machine Interface: A look in to the Future

The drive towards e-Manufacturing and focus on Manufacturing Execution Systems (MES) provides would be the next step ahead in the evolution of HMI systems. The HMI will not be asked to perform MES or Enterprise Resource Planning (ERP) applications, but support integration into these applications. Some near term requirements of HMIs will be transforming portions of its applications into smaller embedded devices, extending HMIs into fully distributable enterprise solutions and better support of mobile corporate personnel through wireless communications.

Traditionally, HMI systems have evolved hand-in-hand with Windows as the underlying operating platform. It would hence be safe to imagine HMIs built on the next generation of Microsoft .NET Framework. In fact, a subset of the .NET Framework is built specifically, to address the problem domain of HMIs.

4.1. The .NET Compact Framework

The .NET Compact Framework is a hardware-independent environment for running programs on resource-constrained computing devices. These devices include mobile phones, personal data assistants (PDAs), set-top boxes, and custom-built embedded appliances. The .NET Compact Framework supports a subset of the .NET Framework class library, and will grow to include classes exclusively designed for the .NET Compact Framework. Here, HMI Panels would essentially be Handheld PC Pro devices without a keyboard, attached to an industrial machine or PLC.

The Future of HMI:
The next generation of HMI systems would leverage the power of Microsoft .NET Compact Framework and enable plant floor machinery to be controlled via mobile phones, PDAs and custom-built embedded appliances.

The capabilities of the .NET Compact Framework would let engineers develop applications on a desktop computer using extensions to Visual Studio .NET, and download the results to an embedded device. Engineers can work in high-level languages like C# and VB .NET and won't necessarily have to write ladder logic. In addition, the .NET Compact Framework gives devices access to back-end servers and Web servers, allowing a lot more flexibility in managing a factory.

Currently, the setup and maintenance of a factory's automation infrastructure can be a nightmare. Windows CE .NET, with its Universal Plug and Play, offers the possibility of actually automating the setup. A new device can be detected and registered automatically, and a failed device can be detected automatically.

5. After Word

One thing is for certain, Human Machine Interface systems will continue to evolve in emerging technologies that support manufacturers' needs, thus bringing additional value to tomorrow's industrial applications.