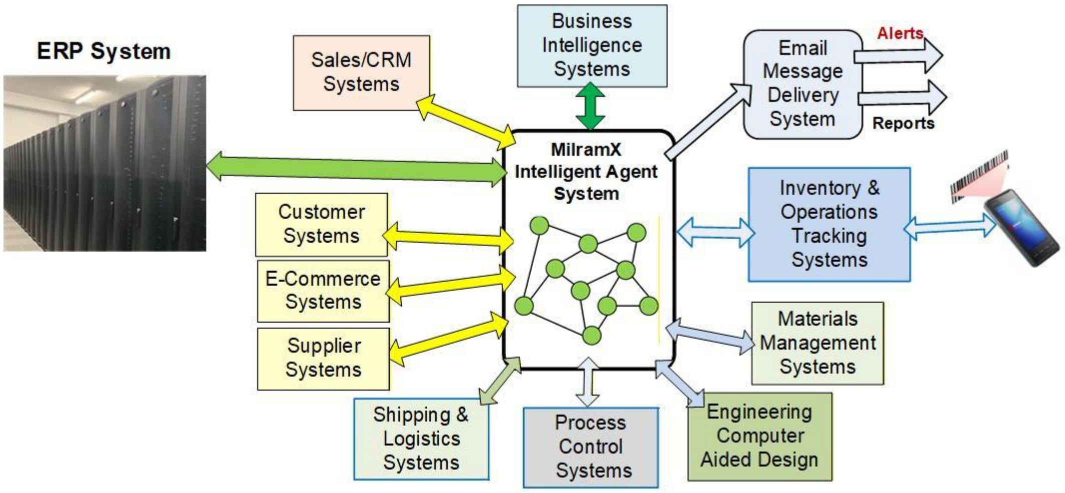
MilramX™ Intelligent Agent Platform Overview


ERP System
Sales/CRM Systems
Business Intelligence Systems
Email Message Delivery System
Alerts
Reports
MilramX Intelligent Agent System
Customer Systems
E-Commerce Systems
Supplier Systems
Inventory & Operations Tracking Systems
Materials Management Systems
Shipping & Logistics Systems
Process Control Systems
Engineering Computer Aided Design

Introduction

MilramX is the metaphorical “Swiss Army Knife” for building automated information exchange applications between multiple systems and the people that use them. Its goal is the enable the rapid and cost-effective implementation of systems that eliminate “intelligent grunt work” by ensuring that all the stake-holders in an industrial organization get the information they need to do their jobs when and where they need it, in a format that is most useful.

MilramX provides typically over 90% of the code needed for these applications, working “out of the box” with the rest provided by Python scripts, which are executed as independently scheduled Intelligent Agent Processes (IAPs). These Python scripts use Tau Adaptor principles to simplify and minimize the number of lines of script code that need to be written for any given application. They make it possible for IT people, business analysts, and manufacturing engineers to configure and create application specific data exchanges without needing to be expert programmers.

The reason that they are called “Intelligent” Agent Processes is that they do not simply move data but analyze the data to extract information, which can be passed directly to people in the form of Emails, or indirectly through the systems that they use. In doing this the agents run 24x7 to replace “intelligent grunt work” tasks which would otherwise be performed by operations managers and their staff.

These intelligent tasks can be as simple as preventing the need for the manual entry of similar, but not identical, data into multiple systems. These tasks can also be as complex as detecting and preventing the use of counterfeit materials or the theft of expensive materials. More typically they are doing tasks such as automatically converting customer orders into manufacturing orders and purchase orders and routing these to appropriate systems and people for action.

MilramX Purpose

Technically, MilramX is a software platform whose purpose is to schedule, run and monitor many Intelligent Agent Processes in real-time, in parallel. These IAPs periodically collect data from one or more sources, such as business systems, databases, and process control systems, synthesize the data into meaningful information, and then use this information to update other systems, or send warning alerts or report to people.

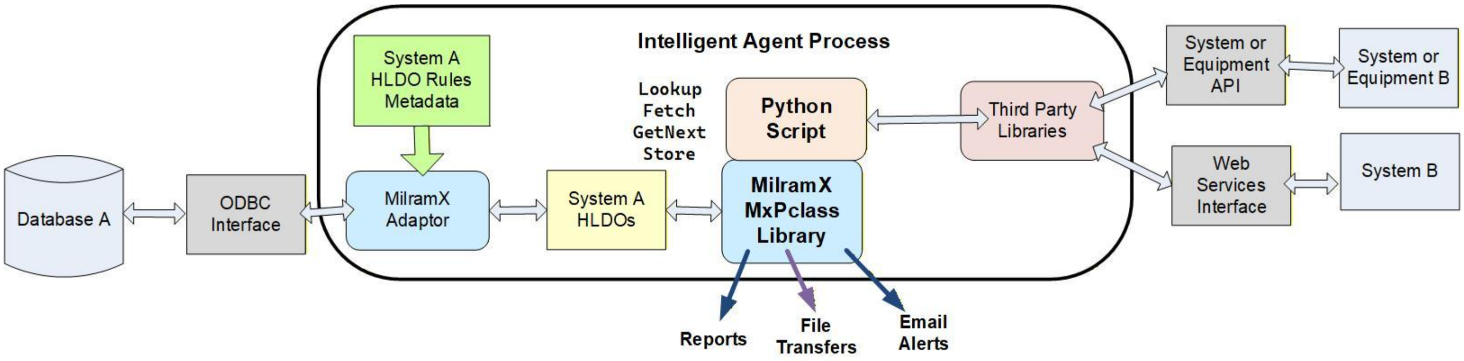
This serves several purposes within an industrial organization, which are:

1. Provide people within an enterprise with the information they need to do their jobs, when they need it, in a format that is easy to use.
2. Avoid the need duplicate manual data entry in multiple systems.
3. Alert people when events occur, or fail to occur, that they need to pay attention to, without the need to continuously monitor multiple sources of data.
4. Automatically generate and Email reports on a periodic basis or when certain events occur.
5. Avoid the need for coordination and planning meetings by automating routine scheduling and planning operations.

The result of this is to save the precious time of operations managers and their staff by automating "intelligent grunt work" which would otherwise take many hours of these people's time. This, in itself, makes these people happier by making their jobs much easier.

From an economic viewpoint this saves the cost of support staff, which would otherwise be required to perform these tasks manually. It also enables the more efficient operation of a manufacturing plant or warehouse by making sure that decisions are made quickly, work is performed in the correct sequence, and customer orders are delivered on time.

Intelligent Agent Processes



Intelligent Agent Processes (IAPs) are executable programs, which are run as background processes on a Windows based computer. They are based on Tau Adaptor principles (see Tau Adaptor User Manual).

Most IAPs consist of a Python script which, when executed, calls upon the MxPclass library, which then dynamically links to all the libraries the IAP needs to access databases, exchange data with remote computers, send Emails and files. This includes linking with third party libraries to exchange data with process control systems and equipment and well as to exchange data with systems that use REST interfaces.

Accessing databases, as well as external systems is simplified by the use of the MxPclass library, which reduces accessing databases to five simplified methods: Lookup, Fetch, GetNext, Store and Run. This makes it possible for IT people, business analysts, manufacturing engineers and other people who are not expert programmers to develop their own IAP control scripts or to easily understand and modify IAP scripts written by other people.

Experience has shown that a typical IAP, such as to move changes in customer data from one system to another, requires less than 100 lines of Python script and, as a result, can be quickly created by someone who is not an expert programmer. Even better the functions performed by an existing IAP can be easily understood and modified if needed by someone who is not a Python programmer.

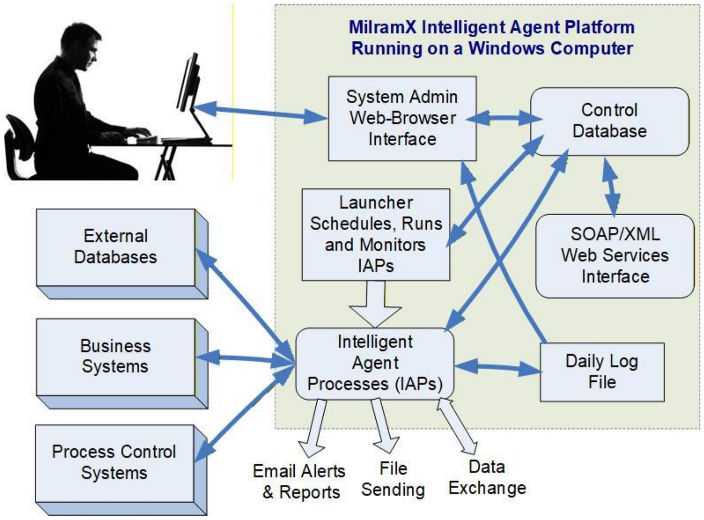
Experience has shown that a typical application may require dozens of IAPs all running independently on different schedules to, for example, exchange data between a barcode tracking system, such as BellHawk, and an ERP system. For this reason, MilramX has a web-browser interface, through which Python scripts can scheduled, run and monitored.

As described in the next section, the web-browser interface interacts with the scheduling mechanism through a Control database. This enables MilramX to also be run “headless” with changes to schedules and the like controlled remotely through the MilramX SOAP/XML web-services interface.

Scheduling IAPs

IAPs are launched and run under the control of a Launcher process, which runs continuously, typically as a Windows Service. This is so that the Launcher will be automatically started in the event of a power failure or a processor reboot.

The Launcher process takes its IAP scheduling information from the MilramX Control database. This specifies how often the IAP is to be run, between which hours, whether it is to retrieve all the specified data available or just the latest updates, and whether the IAP is to be run once or periodically


MilramX Intelligent Agent Platform
Running on a Windows Computer
System Admin Web-Browser Interface
Control Database
Launcher Schedules, Runs and Monitors IAPs
SOAP/XML Web Services Interface
External Databases
Business Systems
Process Control Systems
Intelligent Agent Processes (IAPs)
Daily Log File
Email Alerts & Reports
File Sending
Data Exchange

MilramX is designed to be run on a multi-thread/multi-core computer, which can support the execution of multiple IAPs in parallel, as well as software such as the IIS webserver and SQL Server. When MilramX is installed and configured, the number of IAPs which can be run at the same time is configured, depending on the number of available cores. This is normally one IAP at a time for 6 cores or less but can extend to running 8 or more IAPs in parallel on multi-core processors, such as Intel i9 and AMD Ryzex 9 processors.

The Launcher runs in a loop, typically with a ten second delay, between loops. On each loop, the Launcher:

1. Monitors running IAPs and terminates any IAP that has run longer than its specified run-time. This us to recover processor resources from IAPs that are hung in a loop or hung trying to read or write using third party libraries, which may have bugs.

2. Determines how many available "slots" (allocated threads) it has available to run new IAPs. If there are one or more slots then it picks the most important IAP (based on its importance level, such as rush, normal, low) that is scheduled to be run.

3. The Launcher then runs the selected IAP by executing the IAP's Python script. This then links in all the needed DLLs (Dynamic Linked Libraries) needed for IAP execution. The Launcher then passes it any needed argument data and sets its Windows process priority to below normal so that its execution will not interfere with other software (such as the BellHawk barcode data collection software) running on the same computer.

4. The Launcher then loops back to step 2 and runs as many IAPs in parallel that it has Slots for, before sleeping for 10 seconds or so and looping back to step 1.

When an IAP finishes its assigned task, such as moving updated data from one system to another, it terminates its own process. Thereby freeing up the computer resources used by the IAP. And, if the IAP does not terminate itself within a designated time, the Launcher terminates the process. This enables MilramX to keep scheduling IAPs 24x7 even though a IAP or its adaptor may hang or not perform correctly.

In addition to the databases or systems that the system is configured to grant access to the IAP, the IAPs have access to the Control database, where additional information, such as character set transformation information is kept. IAPs also have access to a daily log file where errors and warnings are logged.

MilramX comes with a web-browser interface through which the scheduling data in the Control file can be updated. This interface also gives access to the status of the IAPs including any error or warning messages they generate.

If the Launcher or any IAP or the DLLs generate error messages these are logged in the daily log file and can also be sent as an Email alert to an IT person. The IT person can then come in remotely through the Sys Admin browser screen and have full access to the log file, the error and warning messages generated, as they will to the databases in HLDO format.

Applications

Some of the applications to which MilramX has been applied include:

1. Implementing data transfers between a BellHawk barcode industrial operations tracking system and a wide variety of ERP and accounting systems.

2. Detecting counterfeit pharmaceuticals before repackaging

3. Detecting theft of precious metals in a production facility
4. Sending food and pharmaceutical traceability records to downstream customers.
5. Automatically detecting and predicting inventory shortages and generating material movement and purchase requisitions in response
6. Automatically converting a flow of customer orders, from multiple sources into manufacturing, picking, packing and shipping orders and routing these to multiple plants.
7. Monitoring production lines in a remote plant in Mexico from a US HQ facility.
8. Taking data from multiple RFID portals and automatically converting this to movement of pallets within an integrated production plant/ warehouse facility.

As well as being run on a stand-alone basis, MilramX is integrated into the following software platforms:

1. WIPtracker - an integrated manufacturing execution and warehouse management system.
2. iMTATs - an intelligent material tracking and traceability system for FDA compliance
3. SmartOps24x7 - which is used to integrate multiple WIPtracker systems, running in plants and warehouses at geographically dispersed locations, into a single cohesive enterprise-wide system.

Commentary

MilramX runs in real-time on inexpensive Windows Server, Workstation, and IOT Enterprise based computers, which can be located on-premises. This is in contrast to regenerative AI (Artificial Intelligence) agent-based systems which require the use of large expensive data centers, accessible only over an unreliable Internet.

Over the past decade, MilramX has proven itself to be able to support all the real-time operations management decision making needed for a modern industrial enterprise, using a network of small inexpensive computers. This is because MilramX uses real-time AI methods developed using funding from DARPA, the USAF, and NASA which was designed to run on small computers located in fighter aircraft, missiles and warships.

These methods include the use of store-and-forward capabilities to help ensure reliable communications even when communications become unreliable, links between computers go down for a while, or some MilramX computers are mounted on mobile platforms which sometimes go out of range, such as at field sites.