



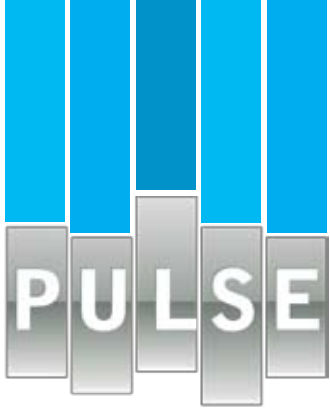
DON'T MISS A BEAT

# AFCON PULSE SCADA/HMI

## Product Description



AFCON SOFTWARE AND ELECTRONICS LTD.



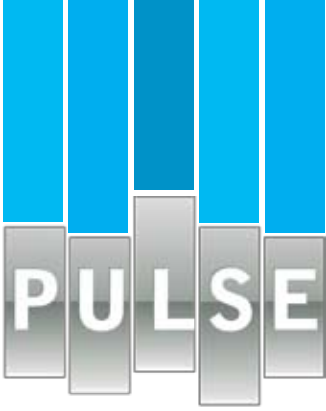
## Introduction

Pulse is a next-generation platform for the development of industrial, building, and security SCADA/HMI applications. Based on the established and successful P-CIM solution, it implements a client-server architecture, incorporating the latest Microsoft technologies such as .NET Framework to bring increased efficiency, superior visualization capabilities, and a highly reliable platform for the supervision and control of manufacturing environments and organizations.

**Pulse offers designers and developers the ability to work autonomously while sharing over the net, to maximize productivity. It makes SCADA/HMI simpler, more scalable, and more cost-effective than ever before.**

Pulse's Client/Server architecture provides flexibility in the deployment and maintenance of SCADA/HMI application across the organization. Pulse uses Microsoft's ClickOnce™ technology, in which the user installs and runs the application by simply clicking a link on a web page, instead of installing entire SCADA systems on every client workstation in the network. Pulse's workstation can run remotely from any location, allowing both operators and application engineers remote access from any location.





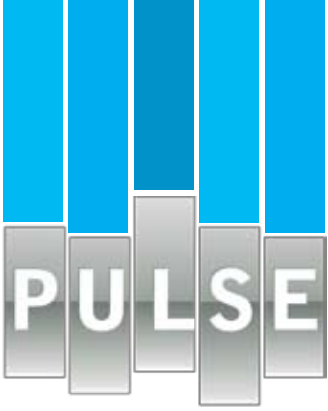
## Sophisticated SCADA/HMI Applications

Pulse offers next-generation capabilities that can be used to realize advanced SCADA/HMI applications. Based on advanced web-friendly architecture, Pulse provides the right information, at the right time, anywhere. Pulse offers sophisticated functionality without compromising on efficiency, intuitiveness and ease of use.

- **Remote Access** - Pulse's runtime environment can be accessed from thin clients through remote terminal servers, or through smart clients using Microsoft's ClickOnce™ technology. ClickOnce enables the user to run a Windows application by clicking a link in a web page, substantially simplifying the deployment and allowing instant remote access from any PC.
- **Rich Realistic Visualization** - Pulse's rich visualization environment offers a realistic representation of the facility and the systems that are monitored. Objects in the facility, such as silos, meters, tanks, pipes etc. are graphically represented with animation, together with switches, selectors and sliders, which imitate the look and feel and operation of the real-life equipment for intuitive operation and monitoring.

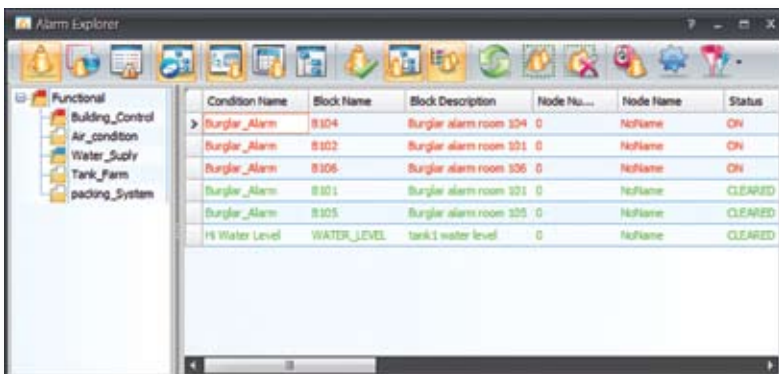


*Rich visualization of the facility and systems*



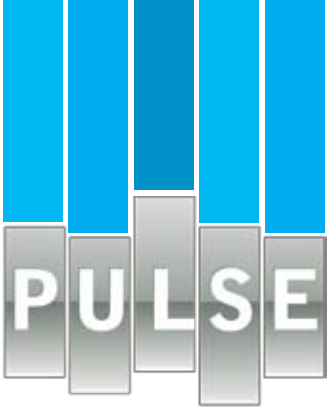
## ►► Sophisticated SCADA/HMI Applications

- **Recipes** – Pulse offers simple management of recipes. The tasks defined in the recipe may be configured to run automatically during runtime, or to be triggered by an operator by clicking an action button. Depending on the task which the operator chooses to carry out, the Pulse Server loads the recipe's data items to the PLC. Conversely, the Server learns data from the PLC. Pulse can learn a new recipe from the controller itself. It allows saving any changes to recipe that were made at the controller level. The load/learn results may change values in the PLC or the recipe file, respectively. Operators can independently create recipes as well as modify existing ones. Due to the sensitive nature of recipes the modification is managed by a sophisticated and granular permissions system, allowing modification of specific subsections of the recipes according to predefined authorization policies.
- **Alarms** – Alarms are displayed in a visible and intuitive way, allowing the operator to quickly understand the alarms' source and how it can be fixed. The alarms are displayed in a hierarchical tree with multiple views, including geographical view and system (functional) view. Alarms are based on granular conditions. The alarms are displayed in an Alarms window, sent by email or SMS. Pulse not only alerts, it provides the means to take action. Alarm can include a link for a certain operation, such as halting a process. User customized Actions can be performed when an alarm occurs, when acknowledged by the user, or when the state returns to normal.



Alarm Explorer





## ►► Sophisticated SCADA/HMI Applications

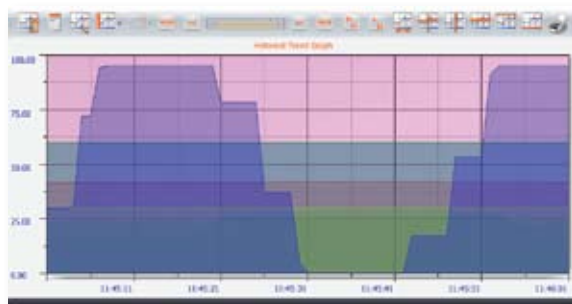
- **Automatic Status Indicators** - Pulse provides status indications (bad, unknown and last good value) for communication between the Pulse server and Pulse publishers (e.g. PLC communication).
- **Pulse Annunciator** – This feature gathers alarms from various displays and presents them in a visual indicator. When the alarm is generated, the indicator changes its colors, indicating a problem in the display or in a process. Clicking the indicator pops up the relevant display.
- **Datasheet** - Compound data from multiple data sources can be displayed in a table format. The table can combine Pulse data with additional data that originates from external systems such as logistics systems, ERP and Maintenance.

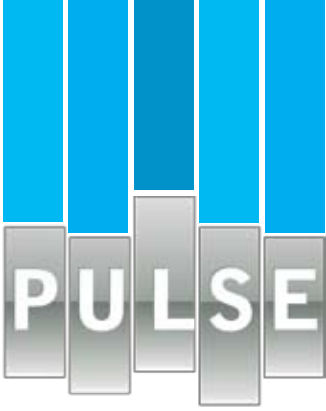
*Datasheet*



- **Trend graphs** – Pulse’s trend graphs display a set of data values in their order of occurrence over a period of time. Trend graphs are used for analyzing processes as they occur in real-time (real-time trends) or during past periods (historical trends). Deviation Graphs indicate, at-a-glance, whether or not each defined advised item is in its normal range or has deviated to unacceptable levels. Curve Trend Graphs are dynamic graphs that present real-time data compared to reference values over a period of time. Batch Graphs display historical data over batch-related time frames. Pulse’s trend graphs can also display several items (more than the conventional eight) at once, over varying time periods.

*Trend Graphs*

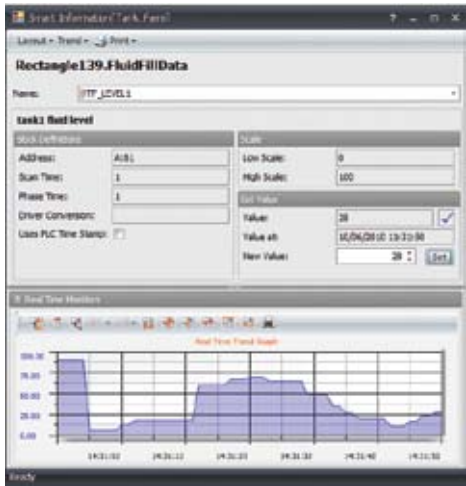




## ►► **Sophisticated SCADA/HMI Applications**

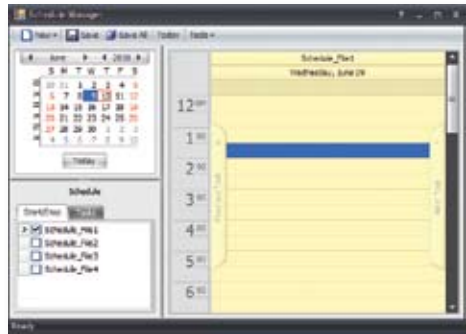
- **Reports** – Pulse’s advanced report and analysis tool, Supreme Report, improves the speed and quality of decision making. It offers the ability to generate reports from real-time, historical and alarms information, collected from multiple data sources. Users can produce scheduled or on-demand reports directly from their Pulse Client workstation. The reports include rich formatting and layout features and can be generated in both PDF or MS Office formats.
- **Smart Information Panel** – Pulse offers easy access to in-depth information on every animated graphic object. A simple right-click on one or more objects opens an information panel that displays information such as, the name of the object, its description, its real address in the field, various SCADA parameters, scales, values, and more. It also allows setting the values, and displays the real-time and historical trends.

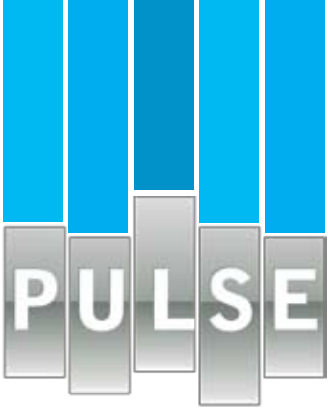
*Smart Information Panel*



- **Scheduler** – Pulse’s Scheduler is an integrated tool that automatically controls the equipment based on calendar events and performs scheduled tasks (actions and workflows). Using a calendar-based user interface, the manager or operator can easily set scheduled actions to any element. The scheduler includes preconfigured holidays, allowing easy setup of operation policies during the holiday periods. Managers/Operators can assign predefined schedules to any data item (e.g. light circuit which is defined in the selected object(s)). Every element can be related to a certain start/stop schedule and any modification in the schedule is automatically applied to all the related objects.

*Scheduler*

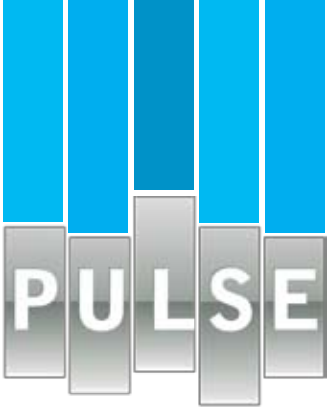




## ►► **Sophisticated SCADA/HMI Applications**

- **Localization** – The Pulse front-end and application interface can be completely localized. All the menus, buttons, screens, dialog boxes, and other interface elements can appear in the local language.
- **Audit Trail** – Pulse records of all the actions and modifications performed by the engineer, allowing maximal traceability in order to comply with legal requirements and regulatory compliance.
- **Logger Policy** - Pulse’s Logger Policy service records events on the system and publish them to one of four default logs/destinations: in the OS Event Viewer, Pulse Logger Window, Excel or as a message box. Event types/sources can be category and their severity can be customized.



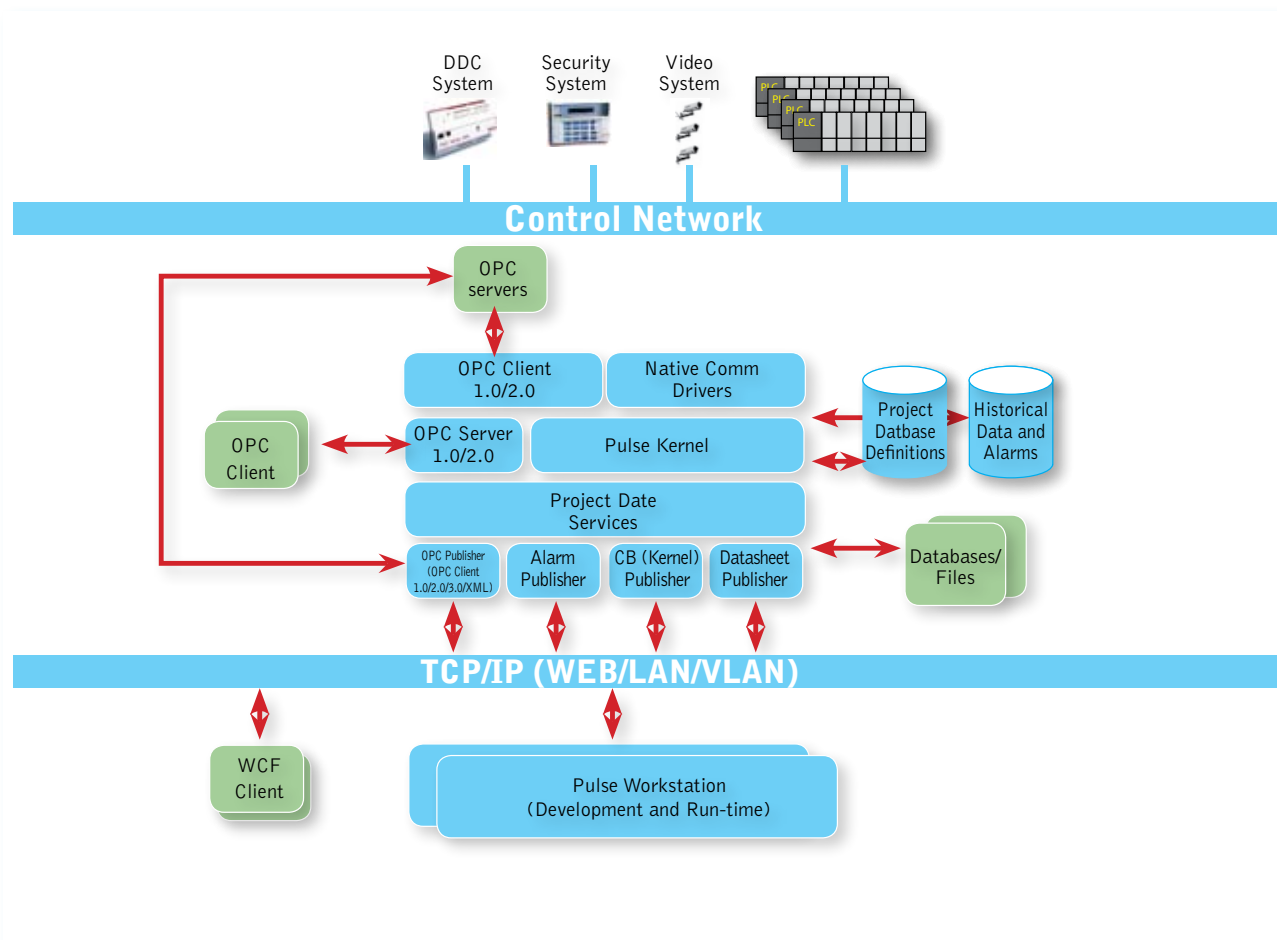


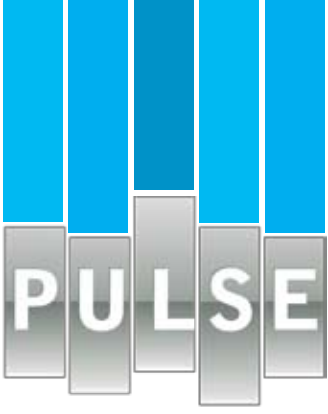
## Architecture

Pulse employs a Client/Server architecture that provides flexibility in its deployment, operation and maintenance. There is no need to install Pulse on every client workstation in the network. Instead, the application and project data resides on the Server, while the operator and developer clients connect to the server through smart clients using Microsoft's ClickOnce™ technology, or through remote terminal servers.

The following diagram describes the Pulse's architecture:

### PULSE ARCHITECTURE





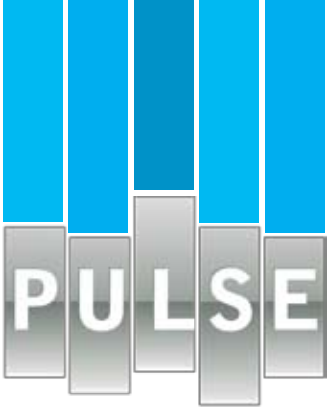
## ►► Architecture

### Inputs and Outputs from Controlled Equipment and Applications

Pulse interfaces with PLCs, automation/control applications, field devices and other types of controlled equipment and applications. Inputs into Pulse can arrive in the following methods:

- **Native Communication Drivers** - The devices communicate directly with Pulse through its native communication drivers. Pulse includes drivers for most major vendors, including Schneider Electric, Siemens, Allen Bradley, GE, ABB, Omron, and many more.
- **OPC** - The OPC (OLE for Process Control) protocol is a set of interfaces that enable standardized data exchange between automation/control sources and applications.
  - **OPC Client** - The OPC Client connects to OPC servers and exchanges data with them, offering a seamless interface with OPC compatible sources. The OPC Data Adaptor explores the OPC Servers that are installed on the server PC (under the 'Local' branch), and its neighborhood (under the 'Network' branch). It gathers additional information about its configuration and actual state (for example, number of connections, advised items, and so on).
  - **OPC Server** - The OPC Server publishes Pulse kernel items, such as database blocks and direct IO addresses, to OPC clients using the P-CIM OPC Gateway service.





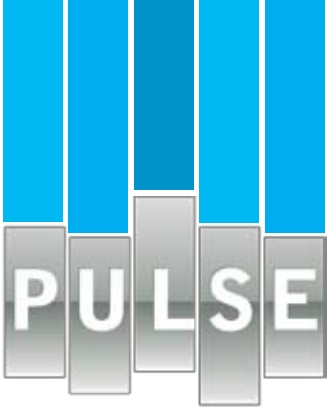
## ►► Architecture

### Pulse Server

Pulse Server receives and sends data to and from the various automation/control applications. After processing the data it disseminates them to the various clients according to predefined rules. The Pulse services can be set to run on different machines. This option is mainly used in distributed applications or large scale projects, when more than one server is needed. For example, one server may be used for the alarms and another for database and historical data acquisition.

- **Pulse Kernel** – The Pulse Kernel is the core of the Pulse server. It manages the drivers, writes historical data and generates alarms. The Pulse Kernel handles communication with the PLCs using P-CIM native drivers and/or OPC servers.
- **Publishers** – Pulse’s publishers are responsible for disseminating data to the various clients. The clients subscribe to the service in order to receive the relevant data according to the permission levels and in order to send control data through the publishers to the controllers. There are five publishers:
  - **Alarm Publisher** - Publishes Pulse kernel-generated alarms and events messages to Pulse clients.
  - **Callback Adaptor Publisher** - Publishes Pulse kernel-generated real-time and historical data to Pulse clients and facilitates the data exchange between the Pulse Kernel and the Pulse Clients.
  - **Datasheet Publisher** - Publishes data from OLEDB/ODBC databases and text files (XML/INI) to Pulse clients.
  - **OPC Adaptor Publisher** – Provides data exchange between OPC DA (Data Access) servers and Pulse clients.
  - **Simulator Publisher** - Simulates and publishes internal items. It manages a few built-in items which frequently change, and can create automatic items, in accordance with application requests.





## ► Architecture

**Database** – Pulse’s database holds the configuration data, the historical data, and the alarms. The default Database is SQL Server 2008 Express, however the user can define any OLEDB database and its location on the network. The database can be exported or imported, allowing for third party software, such as MS Access, to edit or modify the data.

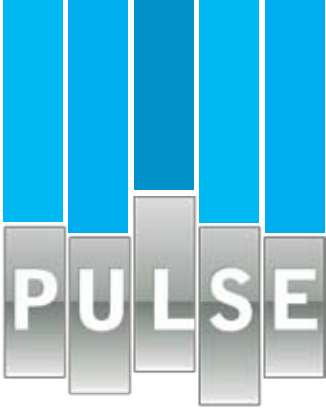
**Message Bus** – Manages the clients’ subscriptions to the various Publishers. This enables the clients to subscribe to the service in order to efficiently receive the relevant data.

### Pulse Workstation

The Pulse Workstation is the user’s (operator and/or developer) interface with their projects. The entire project’s files remain stored on the Pulse server and the remote client user logs into the project over an Internet or Intranet connection through a web browser. Multiple client workstations, each connected to a different Pulse server (Master Host), can run on the same computer. A client workstation can be activated from the local host computer’s web browser.

A client workstation can be also activated from a flash drive connected to one of the computer’s USB ports.





## Development Environment

Pulse presents a new approach to the development of SCADA/HMI applications. It offers an Integrated Development Environment (IDE) that consolidates all the various tools and functions in a easy to use Windows-based UI. Its fully customized interface includes the most advanced features to facilitate development and accommodate any specific need, including Ribbon menus, custom shortcuts and quick menus, hierarchical project tree, and more. Pulse's client-server architecture supports collaborative development between multiple developers simultaneously.

### Displays

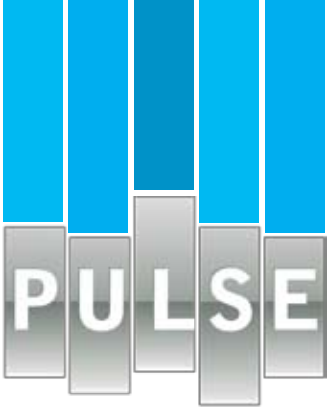
Pulse intuitive development approach is based on displays, a work area that functions as an interactive model of a factory's or organization's facilities (such as a production floor, warehouse, or building lobby). The developers create a collection of dynamic and/or static objects that represent devices or other equipment. The objects have defined properties, which affect how the objects appear and behave according to the data received from the various data sources.

### Graphic Object Properties

Pulse graphic objects include different types of properties, which can be defined by the developer. To facilitate this process, Pulse allows for multiple objects to be selected and edited at once. In addition properties can be copied from one object and applied on another.

- **Advanced Properties** – The properties of all graphic objects and controls can be accessed and edited through the Advanced Properties Panel (Data Sheet control). These properties include size, color, position, orientation, and more.
- **Dynamic Properties** - Pulse instruments and .Net Controls objects properties, such as size, color, location, orientation, font, border style etc., can be accessed/changed during run-time (set property value).
- **Animation Properties** - the animation properties can be accessed/changed using animation tools. These properties determine how each component will be displayed and how they will behave - color size fluid fill ,fluid flow, visibility etc.





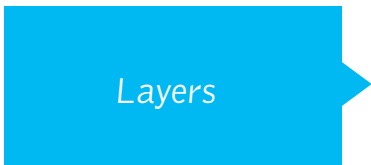
## ► Development Environment

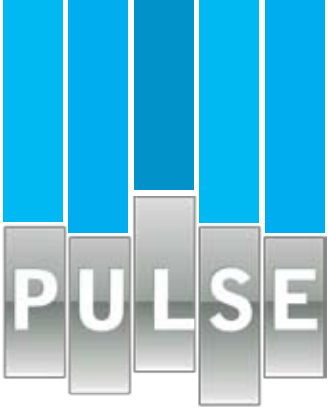
### Data Sheet

Pulse offers the ability to easily display compound data from data sources with its new and advanced Data Sheet (Grid) feature. It enables connection to a database and display the database data in a table format. The Data Sheets support complex data grids, such as OLEDB, XML, Dataset, INI, Text and more. The displayed data can be filtered. The data or any subset (e.g. row) can be transferred to Pulse items by simply clicking it.

### Layers

Pulse's features the ability to separate the various disciplines and systems in different layers (e.g. Electricity, Piping, etc.). Any object of a display can be assigned to layers and thereafter certain layers may be shown or hidden at the workstation. This feature is useful during operation and development, as developers can show or hide each layer in order to manipulate elements in one layer without affecting the appearance or location of elements, which are assigned to hidden layers. Each layer can be assigned with usability and visibility security settings. When applied, users are able to see and use only the layers for which they are permitted in according to their access levels permit. The Layers can be defined during development, through the Layers Editor as well as during runtime.





## ► Development Environment

### Expression Solver

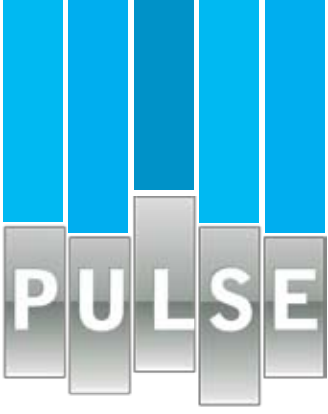
An expression is a mathematical, logical or textual formula which is calculated during runtime and whose result is used as input in an item address or action command. Pulse features a tool for testing the validity of these expressions. checks the syntax of configuration items, variables, tags, functions, aliases, and other functions, when a user writes them into Item's addresses.

### Workflows

A workflow is a collection of activities which Pulse executes in a defined sequence. The workflow is presented to the developer in a flow chart allowing simple and intuitive configuration. Workflows can be triggered by the following events:

- Activating an object by Action Button Animation
- Activating a Pulse Action Button Instrument
- Executing a workflow from the Workflow Explorer
- Display Startup action
- Display Shutdown action
- Display AutoAction
- Workstation startup action
- Workstation shutdown action
- Workstation on - idle action
- Workstation AutoActions
- Scheduled actions and triggered server actions
- .NET Control dynamic events





## ► Development Environment

### Shortcuts

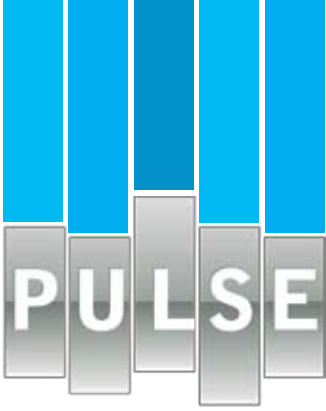
Pulse allows easy access to commonly used tasks with a combination of customized toolbars and context sensitive right-click menus. Developers can configure and customize this features to suit their specific needs.

- **Toolbars** - The Toolbar is used to provide the operator immediate and intuitive access tooften-used Pulse Actions - outside the graphics display area. Developers can create customized toolbars separately from, or in addition to their respective graphic displays. Pulse provides unlimited number of customized Toolbars consisting of tooltips and dynamically changed images according to Pulse items data.
- **Right - Click Menus** – right-click context menus (menus that open when you right - click the mouse on a selected object) that can be assigned as an animation property to graphic objects. For example, A graphic object may contain a context menu whose commands vary depending if an alarm condition occurs or returns to normal.

### Pulse AutoActions

Pulse AutoActions offers the ability to set Pulse to perform a variety of functions automatically without the user's interaction. For example, setting Pulse to send an email, load/learn recipes, print reports and switch to a specified data Layer at a certain time period or when a PLC data trigger is invoked. Actions can be set to occur once after a set period of time, or to reoccur on an ongoing basis. AutoActions can defined as "global" and therefore run on the server side (server autoactions), local (workstation) at a specific workstation level, or even at a specific display level. Server-side AutoActions can be configured to run on a "triggered event".





## ►► Development Environment

Workstation-side AutoActions can be configured by the user to run at the following events and as long as the workstation is running:

- Startup Action - triggered when starting the Workstation.
- Shutdown Action - triggered when shutting down the Workstation.
- On Idle – triggered when the system on idle for a configurable period of time.
- On Idle end - triggered when the system returns from idle state to normal state.
- On Triggered event as long as the workstation is running.

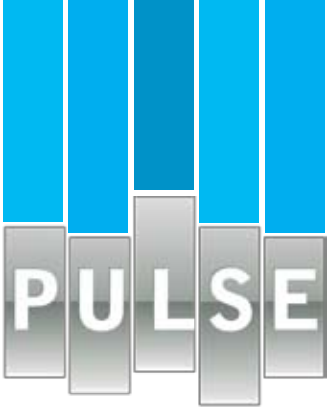
Display-specific AutoActions can be configured by the user to run at the following events:

- Startup Action - triggered when opening the display.
- Shutdown Action - triggered when closing the display.
- On Triggered event as long as the display is open.

### Reuse and Standardization

The key to successful and efficient development is the ability to reuse and repurpose previous developments. Another key to success is ability to ensure consistency and standardization throughout the project, especially when multiple developers are working simultaneously. Pulse excels in this area, featuring multiple solutions that ensure both reuse and standardization throughout the project and between multiple projects:

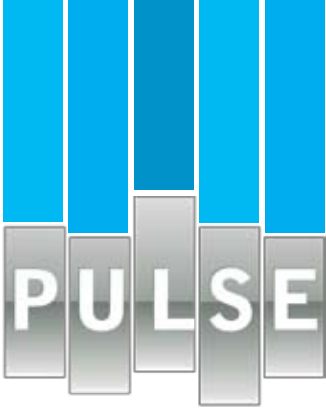




## ► Development Environment

- **Templates** – A template is a starting point for building displays in the Developer Workstation. Developers can populate templates with defined graphic objects, animation properties, auto-actions, workflows, and so on. Building displays from templates saves much time and labor when designing displays for similar production lines or floor layouts. In addition, the use of templates assists in the standardization of the project. For example, using consistent elements such as status bar, menus etc.
- **Themes and Styles** – A theme is a customized set of visual parameters, called styles, that can be applied to graphic object. A theme can have more than one style. A style is a collection of defined graphic properties Object Fill, Object Line, and Text Appearance (includes text, line, fill, font, position, and more). Applying styles to graphic object facilitates the reuse of the same graphic properties multiple times without needing to repeatedly define the properties one at a time.
- **Cells** – A cell is a collection of static/animated graphic objects and clipart objects that are grouped together into a single entity and saved for reuse. The creation of cells can save the repetitive work required to configure several sets of similar object groups. Cells are referenced. This means that any changes to a cell in the Cells Library will apply to all its instances in the application.
- **Pre-defined Objects** - Displays can be drawn from scratch or alternatively existing object that are part of the software can be used as a starting point. Pulse includes the following pre-defined objects and resources:
  - Graphical Objects** - Primitive graphical objects such as rectangles, ellipses, poly-line, polygon, spline, closed-spline, pie, text objects and more.
  - Pulse Instruments** - these include objects such as trends, alarms explorer, buttons, lists, toolbars, slider, cylinder, datasheet, and more.
  - Clip Art** – A library of graphical objects grouped by categories. New objects and categories can be defined by the user.



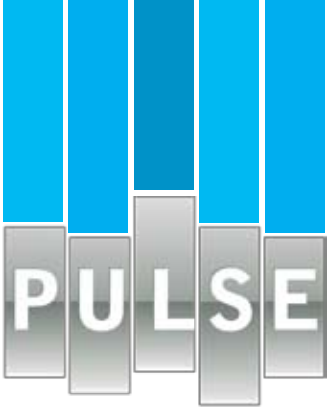


## ►► Development Environment

- **.Net Framework Controls** - .Net framework components are unique objects that perform a task or display information and come with settings that control its behavior, functionality and appearance. These controls and components can be embedded in a display. Pulse provides a variety of .Net controls such as Display, Light, Panel, Web browser, Tab, Calendar, List view, List Tab, DateTime picker, Combo box, Check and Radio button and more. Pulse includes a Controls library with thousands of .Net Controls for industrial automation.
- **Resources** - A Resource is a generic object that represents multiple similar devices or a group of objects. It is a template that enables the reuse of an existing display, while modifying item addresses, without recreating similar displays or copying several objects from one display to another. For example, a resource can be used to represent separate ingredient containers in a food production line, with each container object in the display receiving quantity data from its own item address. Display resources reduce the number of displays needed to store in the project, help save time during project development, and make it easier and more efficient to maintain the project.
- **Aliases** - Aliases are "free text" names that are used as shortcuts for addresses, references, and actions. Association with an alias means applying the change only in the Alias Manager instead of on the related objects themselves, simplifying the maintenance of the application

**Extensibility** - Pulse leverages .Net technology capabilities, by granting advanced users the option to extend the collection of product functionalities with software components they develop themselves. Components such as activities, engineering unit's conversions, .Net controls and more, can be added to the product tools galleries and be reused in the application.





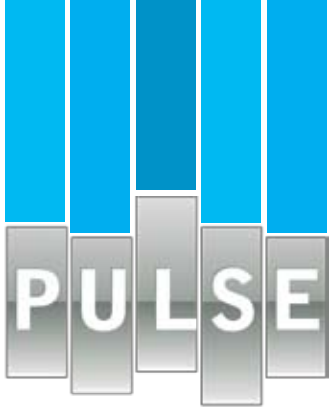
## ►► Development Environment

### Diagnostic Tools

Pulse features several tools to make sure that the project's data and expressions are valid:

- **Data Scope** - DataScope is a diagnostic tool for testing the validity of item values. The tool allows users to immediately receive the current value of any item and then update its values accordingly.
- **Item Explorer** - Pulse's Item Explorer presents the various Publisher items in a hierarchic tree/branch format, allowing users to monitor the status of the Pulse Host's items. Item values can be monitored and saved as Data Scope groups.





## Deployment

Deploying Pulse across the network is simple. In stand-alone mode, the Pulse client workstation and server run on the same computer. In a network configuration, the client/server architecture enables new client workstations to connect automatically to a running server through a web browser, eliminating the need for any local installation. The addition of clients does not require manual modifications to the project files on the Pulse server.

Pulse takes advantage of Microsoft's ClickOnce™ technology to provide a portal for easy access to client connection and version upgrading. Any change in the Pulse version on the Pulse server is automatically updated during the next client connection.

### Hot Redundancy

Pulse supports the addition of a secondary Pulse server in the event that the primary server fails. It includes a built-in hot swap feature that allows the secondary server to automatically take over in the event of failure.

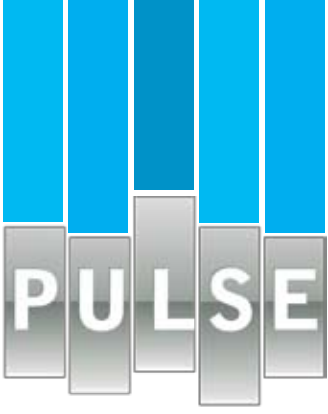
### Backup and Restore

Pulse features a backup and restore wizard that facilitates the process of backing up the data and restoring it on another machine. The result of the backup process is an installation package, which can be password-protected. Backup and restore can be used to protect the data from any damage, to store various stages of the project's development, or to transfer a project to another computer.

### Running Pulse from Flash Disks

A client workstation can be also activated from a Flash drive connected to one of the computer's USB ports for maximum portability. The client is hosted by any available PC without any installation or retaining application data on the hosting PC.





## Security

Pulse provides a selection of integrated security models and options that protect the data from unauthorized access and modifications. This set of mechanisms helps prevent damage and loss to projects as a result of intrusions and configuration changes by persons who are otherwise not allowed to control the system.

Using the Security Manager interface, a Security Policy is defined and assigned to a Pulse project. The Security Policy includes Simple and Advanced Authorization modes, various user authentication modes, access control to modules and menus, audit trails, and I/O write security.

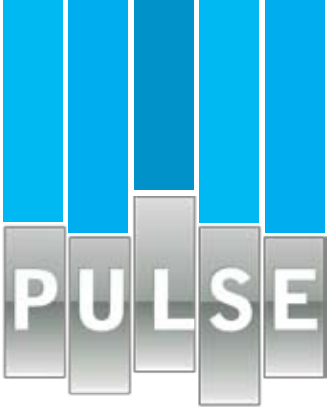
### Authorization Levels

Pulse users can be classified as Administrators, Operators, Developers, Managers, etc., or any combination of these roles, by assigning them an authorization level (Access Level in the Simple Authorization Mode or a Security Key in the Advanced Authorization Mode). According to their authorization levels that can gain access to view (visibility) and/or use (usability) the data.

There are two authorization options:

- **Simple** - Each user is assigned an access level score. Each item, action, or element for which security is enabled is assigned an access level score. Only users with matching access levels or higher can gain access.
- **Advanced** – a multi-dimensional security mechanism, it manages security locks and keys. Every item, action, or element is assigned a lock, which specifies its minimum permission requirements to gain access. On the other end, every user is assigned a key, which specifies the user's permissions. To open the lock, the user must own a key that matches its settings.





## ►► Security

### Authentication Methods

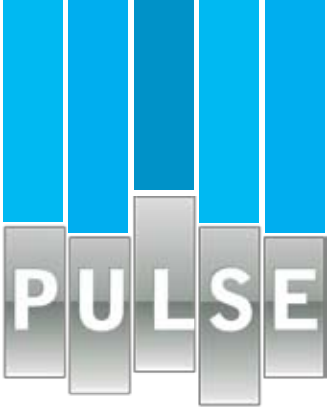
Pulse supports four types of authentication methods:

- **Open (Anonymous)** - this authorization method defines the application as open to every user and the user is defined as Anonymous. When the user is anonymous, the user's permissions are not validated before performing operations.
- **User Management** – management of different authorization levels through the Users Manager. All the security credentials and settings are saved on the Pulse Server in a per-project basis.
- **Active Directory** - Pulse supports Active Directory authentication. The user enters the user name and password into a login page. The password is validated by Active Directory and the user name and authorization privileges are saved on the Pulse Security database on a per-project basis.
- **Windows Login** – According to this method there is no need to perform a separate manual log-in when accessing Pulse. Instead, the authentication is performed automatically with the current Windows user login.

### File Access Security

Pulse features a security mechanism for the protection of the project's files from unauthorized access. This measure prevents third party software from accessing the data.





## Licensing

Pulse's system licensing is extremely flexible, offering the ability to pay exactly for the actual need. The licensing is based on a combination of operation modes and I/O volume:

- **Operation Modes** - The Pulse Client is available in three basic configurations:
  - **Run Time Only:** used by the operators accessing Pulse for monitoring and control.
  - **Development Only:** used by administrators and engineers to create, change, and edit the application.
  - **Viewer Only:** used by users who only need to view, but not control, the Pulse application.

Any combination of client's configuration can be purchased.

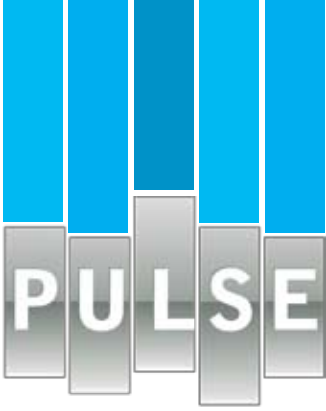
- **I/O Volume** - The I/O count is cumulative, and counted by the Pulse kernel. Unadvised items are not subtracted under the license. An I/O item is an item whose source is a communication driver or an OPC server. Flexible amount of I/O items may be purchased from the following volume: 100, 300, 500, 800, 1500, 3000, 5000 and Unlimited (65000)

### Floating Client Licenses

Floating licenses means that a certain amount of licenses may be shared among a larger number of users over time. These licenses are intended for users who need to occasionally use a client as an operator. Connections will be allowed provided that the number of valid licenses is not exceeded. Floating Client licenses are stored on the key that is plugged into the Server.

For each computer logging into Pulse, a single license is used for each different working mode. Up to three licenses (1 Development, 1 Monitor & Control, 1 Monitor) may be used at a time on a single computer (counted by a Pulse server). If a developer user and a monitor user both log in from the same computer, Pulse records 2 licenses used.





## Software Requirements

Operating System	Pulse Components		
	Pulse Server	Pulse Client	Standalone
Windows 7 Ultimate	No	Yes	Yes
Windows Vista Enterprise	No	Yes	Yes
Windows Vista Ultimate	No	Yes	Yes
Windows Server 2003 Enterprise SP2	Yes	Yes	Yes
Windows Server 2003 standard R2 SP2	Yes	Yes	Yes
Windows Server 2003 Enterprise R2 SP2	Yes	Yes	Yes
Windows Server 2008/ R2	Yes	Yes	Yes
Windows XP Professional SP2	No	Yes	Yes
Windows XP Home Edition SP2	No	Yes	Yes

