10 KEY FEATURES OF AN AUTOMATION CHANGE MANAGEMENT SYSTEM

Programmable automation devices (PLC, CNC, SCADA Systems, robots, etc.) and their logic programs are vital to the running of a plant and adjustments to variables and logic are needed to continue smooth operation. If program changes are not managed correctly and the current device program and configuration are lost, an old version of the device program must be used, resulting in decreased machine performance, decreased quality, increased safety hazards and downtime.

An Automation Change Management System (CMS) is a centralized system that manages changes to program logic. The selection of the best CMS application to meet a facility's needs requires a careful assessment of the features provided by a CMS and the plant's requirements. It is recommended to consider the following:

1 An archive of prior versions of programs

Many facilities will set retention parameters to maintain a representative number of prior versions of program files. These parameters will include the number of copies to maintain and a minimum age of a copy to be deleted. The age requirement is very useful when multiple unsuccessful attempts are made to correct a program issue and it is determined that reverting to an older copy of the program would be a better starting point than recent edited versions.

2 The ability to detect and notify users of program changes

Optimal CMS benefit is achieved when users go through the CMS to make all program changes. This ensures a complete history of changes. The CMS should also have the ability to interrogate devices and compare the program running in the device to a reference copy in the CMS. If changes are detected, appropriate identification and notification should occur.

3 Tools for documenting changes and making them visible to users

As program editor software packages vary in their capability to identify changes, the CMS yields tremendous benefit by providing a consistent, intuitive tool to compare changes between any two versions of a program. This could be between a master copy, prior version or the current version in the processor.

4 Secured user and workstation access

Each user should be authenticated by the CMS. In addition, some facilities have line-of-sight or other restrictions on which workstations can be used to edit certain device programs. This is often utilized when improper program changes could have safety impacts.

10 KEY FEATURES OF AN AUTOMATION CHANGE MANAGEMENT SYSTEM

5

Features mapped to user permissions

Users should be assigned to groups with permission profiles that map to the user's authority within the plant. Careful consideration of these roles during implementation is encouraged. It is also advisable to keep the role structure as simple as possible.

6

Procedures for recovering from hardware failures

If the device hardware fails a replacement device will need to be obtained and connected to the network. A CMS user should then be able to download the latest copy of the program to the device to resume operation.

7

Open standards and architecture

The CMS should use standard hardware and software, not proprietary or open-source platforms or tools which can have licensing or security concerns. This would include modern Windows server and client software support as well as a standard SQL database for ease of corporate reporting and integration.

8

Ease of upgrade and enhancement

As there are frequent updates to the automation software used for plant devices, the CMS should have a modular architecture supporting the ability to easily update and deploy new drivers supporting these updates.

9

Non-networked device management

Disconnected devices should be supported by a checkout and check-in procedure, or if provided by the CMS application, non-networked tools can be used to download copies of programs from the central repository to the controls engineer's laptop for access remotely. It is also helpful if the CMS can compare the program in the device with the one downloaded from the repository, capture user comments regarding program changes, assist in creating new programs in the field, and synchronize all changes back to repository.

10

Plant-Wide Control

For a CMS to be effective in safeguarding a plant, it must support a wide range of devices, not just a few brands of devices, as well as any PC-based application. The CMS should also have customizable and extensible tools for managing all files and data in the plant