3.4 DCS and SCADA/PLC comparison

Manufacturing industries either have DCS or SCADA/PLC systems. With the advances in technology and the resulting overlapping functionality, these systems became less and less distinct. The following system characteristics give some relevant features of the average systems.

DCS	SCADA/PLC
Expensive hardware but engineering is comparatively cheaper	200% cheaper than DCS Hardware is cheaper but engineering is expensive
Control and monitoring over small areas e.g. a process unit	Over large geographical areas
Data transfer via LAN infrastructure	Use leased telephone lines or radios
Large, extensive applications with many control and data transfer Analog control processing	Small medium-sized applications with majority open/close control
Direct control, output directly to field actuators	Send set points to local controllers
DCS was a single vendor solution	With SCADA the connection with the field is done by third party hardware and software
Focused in process industry	Focused in discrete production industry
Inter-related continuous complex processes	Batch processing with low level of process interaction
Robust hardware and software are the part of the equipment on the shop floor and are fixed	Operating system software and the processor used in a SCADA PC undergo quick changes due to heavy competition
Process computer on the shop floor and PCs in the control room	PLC platform for process control and a PC platform (SCADA) for display
Data acquisition is event driven, rely on change. The RTUs can typically operate for extended periods of time w/o communications with the 'Host'	Data acquisition via one database with fixed scan cycle for each data point
Application stored on one database	Application data are divided over several databases
Fast with complex control	Fast when used in logical (on/off) application
Predictable, real time	Not completely predictable
Handle many controls	Limited number of controls

The goals of DCS and SCADA are quite different. It is possible for a single system to be capable of performing both DCS and SCADA functions, but few have been designed with this in mind, and therefore they usually fall short somewhere. It has become common for DCS vendors to think they can do SCADA because the system specifications seem so similar, but a few requirements paragraphs about data availability and update processing separate a viable SCADA system from one that would work OK but for the real world getting in the way.

The DCS is process oriented; it looks at the controlled process (the chemical process plant) as the center of focus, and it presents data to operators as part of its job. A DCS operator station is normally intimately connected with its I/O (through local wiring, fieldbus, networks, etc.). When the DCS operator wants to see information, he usually makes a request directly to the field I/O and gets a response. Field events can directly interrupt the system and advise the operator. The DCS is always connected to its data source; so it does not need to maintain a database of 'current values'. Redundancy is usually handled by parallel processing.

The SCADA is data-gathering oriented; the control center and operators are the center of focus. The remote equipment is merely there to collect the data, though it may also do some very complex process control. The SCADA must still operate when field communications have failed. The 'quality' of the data shown to the operator is an important facet of SCADA system operation. The SCADA systems often provide special 'event' processing mechanisms to handle conditions that occur between data acquisition periods.

These underlying differences prompt a series of design decisions that require a great deal more complexity in a SCADA system database and data-gathering system than is usually found in DCS. The DCS systems typically have correspondingly more complexity in their process-control functionality. The SCADA database architecture is significantly different from the DCS data architecture. The SCADA system is event driven, while DCS is process state driven.