SECTION D

FACILITY MANAGMENT:

Facility management

- The Wonderware Industry Application for Facilities Management provides an open and standardized Supervisory Control and Data Acquisition (SCADA) system for facilities management solutions, able to collect and process building automation information from large distributed campuses or multiple buildings under central supervision.

Facility management

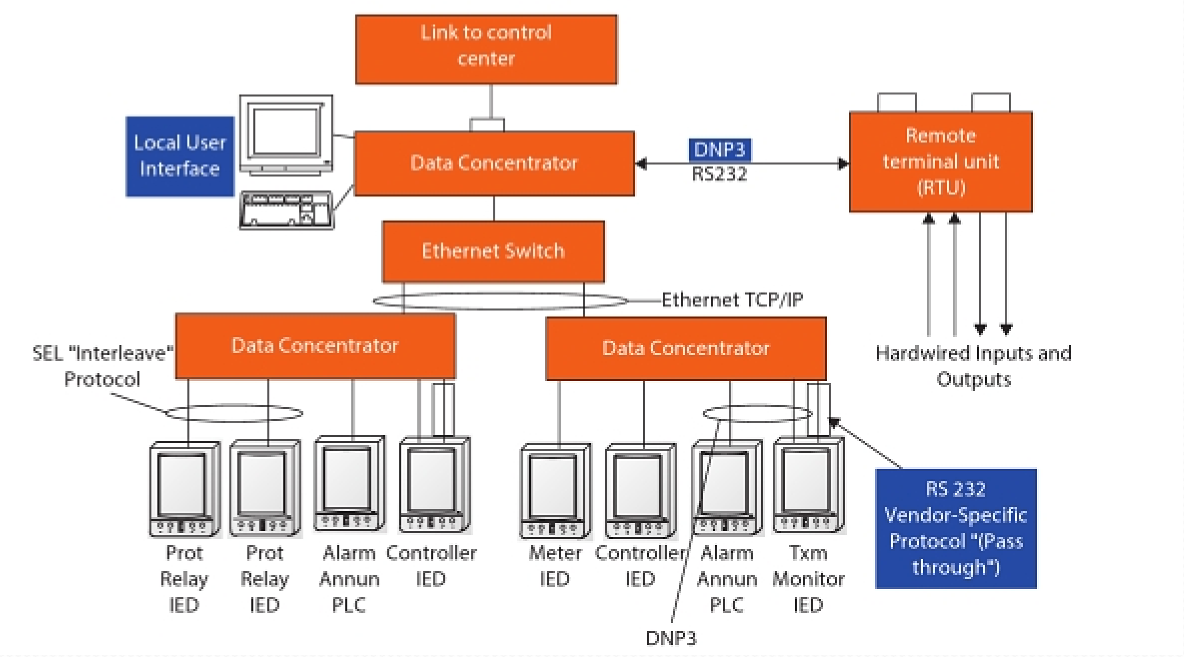
Facility management of SCADA means where SCADA is used.

- **Manufacturing:** SCADA systems manage parts inventories for JIT manufacturing. They also regulate industrial automation and robots. To ensure good output, they monitor process and quality control.
- **Buildings, facilities and environments:** Facility managers use SCADA to control devices. These include HVAC, refrigeration units, lighting and entry systems.
- **Electric power generation, transmission, and distribution:** Electric utilities use SCADA systems to detect two key things. Those things are current flow and line voltage. They monitor the operation of circuit breakers. They also take sections of the power grid online or offline.
- **Water and sewage:** State and municipal water utilities use SCADA to monitor and regulate water flow. They also track reservoir levels and pipe pressure.
- **Mass transit:** Transit authorities use SCADA to regulate electricity to subways, trams and trolley buses. They also automate traffic signals for rail systems. They can track and locate trains and buses with SCADA. They can also control railroad-crossing gates.

Key benefits

- Lower project cost through open, competitive choice for products, services and maintenance
- Lower costs for connectivity and integration by accessing the broadest range of protocols and devices
- Reduce risk with an industry proven robust, secure architecture
- Improve productivity and maintenance by deploying a standards based solution
- Improve efficiency with a library of proven, standards based, optimized applications
- Reduce implementation costs by leveraging the largest global network of independent System Integrators
- Achieve environmental sustainability through compliance with industry, regulatory and green initiatives
- Reduce total cost of ownership by implementing a sustainable solution through the Wonderware ecosystem of partners

Automated mapping



Major Design Issues

1. Object Models and Naming Schemes
2. Distributed Coordination
3. Interprocess Communication
4. Distributed Resources
5. Fault Tolerance and Security

2. Distributed Coordination

- Interacting concurrent processes require coordination to achieve synchronization.

- Types of Synchronization Requirements:
  - In general there are three types of synchronization requirements:

1. Barrier Synchronization
   - A set of processes or events must reach a common synchronization point before they can continue
2. Condition coordination
   - A process or event must wait for a condition that will be set asynchronously by other interacting processes to maintain some ordering of execution
3. Mutual Exclusion
   - Concurrent processes must have mutual exclusion when accessing a critical shared resource

Design Requirements

- Performance Issues
  - Responsiveness
  - Throughput
  - Load Balancing
- Quality of Service
  - Reliability
  - Security
  - Performance
- Dependability
  - Correctness
  - Security
  - Fault tolerance

Hardware

- A single CPU with one control unit.
- A single CPU with multiple ALUs (arithmetic and logic units).There is only one control unit.
- Separate specialized functional units, such as one CPU with one floating-point co-processor.
- Multiprocessors with multiple CPUs but only one single I/O system and one global memory.
- Multicomputers with multiple CPUs, multiple I/O systems and local memories.

Control

- Single fixed control point. Note that physically the system may or may not have multiple CPUs.
- Single dynamic control point. In multiple CPU cases the controller changes from time to time among CPUs.
- A fixed master/slave structure. For example, in a system with one CPU and one co-processor, the CPU is a fixed master and the co-processor is a fixed slave.
- A dynamic master/slave structure. The role of master/slave is modifiable by software.
- Multiple homogeneous control points where copies of the same controller are used.
- Multiple heterogeneous control points where different controllers are used.

Data

- Centralized databases with a single copy of both files and directory.
- Distributed files with a single centralized directory and no local directory.
- Replicated database with a copy of files and a directory at each site.
- Partitioned database with a master that keeps a complete duplicate copy of all files.
- Partitioned database with a master that keeps only a complete directory.
- Partitioned database with no master file or directory.

Network Systems

- Performance scales on **throughput** (transaction response time or number of transactions per second) versus **load**.
- Work on burst mode.
- Suitable for small transaction-oriented programs (collections of small, quick, distributed **applets**).
- Handle uncoordinated processes.

Parallel Systems

- Performance scales on **elapsed execution times** versus number of processors (subject to either Amdahl or Gustafson law).
- Works on bulk mode.
- Suitable for numerical applications (such as SIMD or SPMD vector and matrix problems).
- Deal with one single application divided into a set of coordinated processes.

Distributed Systems

A compromise of network and parallel systems.

Comparison

Item	Network sys.	Distributed sys.	Multiprocessors
Like a virtual uniprocessor	No	Yes	Yes
Run the same operating system	No	Yes	Yes
Copies of the operating system	N copies	N copies	1 copy
Means of communication	Shared files	Messages	Shared files
Agreed up network protocols?	Yes	Yes	No
A single run queue	No	Yes	Yes
Well defined file sharing	Usually no	Yes	Yes

Comparison of three different systems.