WinSEM

OLE Based Real-Time Device Interface (Proposed)

03/24/95

Dan Mitchell
USDATA
dan_mitchell@usdata-richardson.ccmail.compuserve.com
(214) 680-9700

Table Of Contents

1. Introduction

Typically, application software is "monolithic" in that all the functions of the application are linked together. Internally, the software may be constructed in a modular fashion, but the end user is not provided with a way to incorporate additional modules into the application. The need for a means to rapidly construct complex, yet flexible, software systems has lead the software industry to move towards a *component software model*.

Component software is built through the combination of independent software components, much like a component stereo system. A component is a reusable piece of software that can be "plugged into" other components from other vendors with relatively little effort. Component software is a form of object-oriented software that is applied at the application level and not just at a programming level.

In order for component software systems to become a reality, a standard interface that defines how components communicate is needed. OLE provides a standard interface as well as the infrastructure needed to build new interfaces.

Currently, each software package for use in industrial and laboratory automation applications implements its own device interface software. This often makes it difficult, if not impossible, to use various packages together. For example, a PLC programming system is incompatible with an MMI from another vendor. Thus, PLC data points that are defined in the programming system must be redefined in the MMI system. Although some effort has been made to integrate the various systems through the use of import/export files, there is no easy way for the user to combine the packages into an easy to use solution.

In addition, there is a great deal of duplication of effort. Device interface software must be implemented uniquely for each vendor's software. With a common interface, different software systems could coexist and share the device interface components.

2. Objective

This specification describes a standard interface for transfer of data with an external device. The types of devices covered are those typically used for scientific, engineering, and manufacturing applications. The specification covers the following areas:

- Symbolic names
- Transfer of data
- Standard data types
- Device interface configuration
- Error and status reporting

2.1 Symbolic Names

A symbolic name provides a device independent way to represent data and device types and addresses. The use of symbolic names to identify devices and data sets provides greater device independence and also allows for better integration of application programs. The same symbolic name can be used in all applications that refer to a specific device or data, thus reducing the effort need to make changes in device or data addressing.

2.2 Transfer of Data

The primary operations on a device are read and write. From the standpoint of the client, a read transfers data from the device to the client while a write transfers data from the client to the device.

Read and write operations may be divided into two primary types: polled and exception. In a polled operation, the client invokes a function on the device each time it wants to read or write data. In an exception operation, the client invokes a function on the device that causes the device to periodically update the client when data changes (or on a regular basis) without further requests from the client.

A polled operation may be further divided into blocking (synchronous) or non-blocking (asynchronous). In a blocking read or write operation, the client makes the request and then waits until the request is complete, an error occurs, or a time-out period expires. In a non-blocking request, the client issues the request and then later receives a notification that the operation has completed. A non-blocking operation is similar to an exception operation with the main difference being that the exception operation provides for continued notification without further requests.

In order to be useful in a real-time application, data transfer must be high-speed. Although the term "high-speed" varies by context, it generally means that data transfer time should be minimal.

2.3 Standard Data Types

Each device may use different binary representations of values. In order for client software to be device independent, a standard list of data types is needed. The device object is responsible for converting data to the standard data types.

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.