

The Windows operating system provides several mechanisms to facilitate communication and data sharing between applications, between devices, and between applications and devices. Interprocess Communication (IPC) is the mechanism for communication and data sharing between applications. Open Platform Communications (OPC) is the mechanism for communicating and data sharing between devices. These technologies use a Client/Server paradigm whereby the Client connects to and requests data or services from a Server, and the Server responds to Client requests. Depending on the circumstances, ExtendSim can act as either a Client or Server application for IPC and OPC communication.

- IPC enables one application to control another and multiple applications to share the same data without interfering with one another. It utilizes a set of programming interfaces, called communication protocols, which allow the coordination of activities among different applications.
- OPC is a set of standards that specify the communication of real-time plant data between control devices from different manufacturers, regardless of the type and source of data. Using DLLs, applications such as ExtendSim can act as an OPC Client to request and receive that data.

As seen on page 2, ExtendSim supports IPC and OPC through user interfaces and programming capabilities that are based on the industry standard technologies shown below. These technologies allow external applications to control and communicate with ExtendSim and vice versa. They also allow ExtendSim to act as an OPC client, receiving real time data.

In addition to IPC and OPC, ExtendSim supports multiple methods for storing and managing data, such as global and dynamic arrays, linked lists, and (most importantly) an internal relational database.

Protocols and methods for ExtendSim IPC and OPC communication

Protocol/Method	Description	Use
COM (Component Object Model)	A set of object-oriented programming technologies and tools for IPC.	Communicate with, exchange data with, or control another application. Link data that, when chosen, automatically starts another application for data editing.
ExtendSim COM Object Model	The objects and methods that ExtendSim exports for Automation, including Execute, Poke, and Request.	Primary method for ExtendSim to be controlled and communicated with as a COM Automation Server.
DLLs (Dynamic-Link Libraries on Windows)	Libraries of routines or functions written in an external language, such as C++. These are the standard methods for linking between the ExtendSim programming language (ModL) and other languages, applications, and devices.	Call an external code segment from a block's code and perform operations. A block can pass data to the DLL, cause the DLL to calculate using that data, and get the results back from the DLL. For example, in ExtendSim, the Rate library blocks communicate with the LPSolve DLL.
COM DLL	A library developed in an external language, such as C++, to create or define Active X objects.	ExtendSim can load up and communicate with a COM DLL or be controlled by a COM DLL. For example, see how ADO is implemented in ExtendSim.
Automation	A subset of COM (previously OLE Automation). Enables applications to expose their unique features (properties, methods, and events) to scripting tools and other applications. The exposed objects are called ActiveX objects.	Access and manipulate (i.e. set properties of or call methods on) shared <i>automation objects</i> that are exported by other applications.
ADO (ActiveX Data Objects)	An Automation protocol for exchanging information with ADO-compliant databases.	Send an entire table of information at one time; high-speed data interchange. An ExtendSim ADO is implemented as a COM DLL for communication with ADO-compliant databases such as Microsoft Access.
FTP (File Transfer Protocol)	Protocol for sending files or data to and from Internet web pages.	Import from, or export to, files located on a server or on the web.
Mail Slots	Functions for inter-computer communication.	An easy way for ExtendSim to send and receive short (<400 byte) messages. Also provides the ability to broadcast messages across all computers in a network domain.
ODBC (Open Database Connectivity)	A standard programming language middleware API for accessing database management systems.	This DBMS legacy technology has been superseded by ADO.

The technologies listed above have been developed as industry standards. You can incorporate these into your ExtendSim models by:

1. Using ExtendSim menu commands and pre-built blocks.
2. Using the ExtendSim File I/O, IPC, OLE/COM and other functions.
3. Using an external language to access ExtendSim Automation protocols or create DLLs.

IPC between external applications and ExtendSim

The table below shows how ExtendSim can communicate with other applications and with ExtendSim models running on other computers, even while the simulation is running. This allows ExtendSim to work on a wide variety of tasks jointly with external applications such as spreadsheets, external databases, word processors, statistics packages, and so forth.

ExtendSim supports IPC technologies through:	Text File	Excel	ADO-Compliant Databases	Other Communication & Control	Notes
Read or Write blocks	X	X			Value and Item libraries.
Data Import Export block	X	X	X	XML files, FTP site	Value library
Command block		X		Excel Macro, COM command	Value library. Select a command to send and choose when it is sent.
ExtendSim Database	X	X		Read, Write, and Data Import Export blocks	Integrated within ExtendSim. See "ExtendSim Database Tutorial and Reference".
ExtendSim DB Add-In		X		External databases	Allows Excel to create, edit, and export an ExtendSim DB file.
OLE Automation - ExtendSim as Server				External languages control ExtendSim	See the Technical Reference and ExtendSim\Examples\How To\Developer Tips\OLE Automation.
IPC functions		X		Other applications	See the Technical Reference.
OLE/COM functions				Other applications, COM DLLs	See the Technical Reference and the example in ModL library > OLE category.
DLL functions		X	X	Other applications; devices & equipment	See the Technical Reference and ExtendSim\Examples\How To\Developer Tips\DLLs.
ADO (user-defined functions)			X		See the Technical Reference.
Mail slot functions				ExtendSim running on other computers	See the Technical Reference.
Internet Access				Web, internet	See the Technical Reference.
ODBC functions			X		See the Technical Reference and the example in ModL Tips library > Input/Output category.

OPC between ExtendSim and external devices and equipment

Open Platform Communications (OPC) is based on Microsoft Windows operating system's OLE, COM, and DCOM technologies. OPC servers provide a method for OPC clients to access data from a process control device such as a PLC or DCS. ExtendSim can act as an OPC client, connecting to and requesting data and services from an OPC Server using Dynamic Link Libraries (DLLs) on Windows. DLLs are segments of code written in a language other than the ExtendSim ModL language, such as C++. A standardized interface provides a method for linking between other languages and ModL. DLLs can also be used to perform complex calculations utilizing specialized hardware.