

README for App-Link™ iMS™ Developer's Control Set Version 1.0

© 1996-1997 Synergy Software Technologies, Inc.

This document includes updates to the documentation provided with the App-Link Internet Messaging System Developer's Control Set 1.0. The information in this document and online Help is more up-to-date than in the Programmer's Guide.

Contents

General Information

What's New in App-Link iMS

Other Updates

- GetComputerNameEx method accepts a new Format option
- Protocol property supports run-time assignment

Structured Message Support Technology

List of Known Windows Affecting App-Link iMS

- GPF under Windows 95 running multiple 16-bit applications
- Windows NT 3.51 hangs unloading 16-bit applications

General Information

The App-Link Internet Messaging System (iMS) provides effortless deployment of communications between applications over the Internet, Intranet, or any TCP/IP network. App-Link iMS will help you enable your applications to the Internet easily and efficiently, without having to be a TCP/IP guru. App-Link is easy to use, and in no time at all you will be an App-Link expert!

What's New in App-Link iMS

The following items run down the major updates of App-Link for iMS:

- Enhanced 16 and 32-bit control set

The App-Link iMS control set integrates local communication with connectivity over the Internet, Intranet, or any TCP/IP network. The control set includes a VBX, and 16-bit and 32-bit OCX controls which can all interoperate with each other.

- New 16 and 32-bit TCP/IP communication servers

Our new iMS servers provide full access to remote applications over a TCP/IP network.

- New technology for encoding/decoding user types in messages

This release features a technology called Structured Message Support, or SMS for short. Using SMS functions, you will be able to encode and decode any series of data items into and out of variable-length strings for use by the App-Link control.

- New Distribution Wizard

The App-Link iMS Distribution wizard is a small setup-like program that automates the process of selecting the required App-Link iMS run-time files. All you need to do is incorporate the selected files into your custom installation program.

- New manual
- Updated examples

Other Updates

The following items run down other updates that were made to the App-Link interface:

- GetComputerNameEx method accepts a new Format option

The GetComputerNameEx method supports an optional parameter that specifies the format of the returned value. The valid options include: 1 - return value as host name, or 2 - return value as host address. If a format is not provided, the method will return the value in host name format. This is consistent with the previous implementation of the method.

Examples:

Host = `skt.GetComputerNameEx(1)` will format the return value as a host name, e.g. HOMESERVER

Host = `skt.GetComputerNameEx(2)` will format the return value as a host address, e.g. 126.134.122.10

- Protocol property supports run-time assignment

The Protocol property interface was changed to support run-time assignment of the property. This allows you to dynamically bind a socket to a protocol during execution, making it easier to deploy applications for multiple environments.

Structured Message Support Technology

This release of App-Link features new technology for encoding and decoding the content of a message. We call this new technology Structured Message Support (SMS). Using SMS functions, you will be able to pack and unpack any series of data items into and out of a variable-length string. The resultant string can be sent and received by an App-Link socket.

SMS is intended as a replacement for the existing `AplkVBTypeToString` and `AplkStringToVBType` functions that are packaged with earlier releases of App-Link. Although we will continue to support these functions, we encourage you to use the new SMS technology because it is more robust, and is designed to be interoperable between 16/32-bit environments.

The following summarizes the SMS function set:

AlsmStoreAll	Converts a user type to a SuperString
AlsmFetchAll	Copies a SuperString to user type members
AlsmStoreItem	Stores a data item into a SuperString
AlsmFetchItem	Retrieves a data item from a SuperString
AlsmIsIndex	Indicates if a SuperString item is an index field
AlsmLenType	Returns the byte length of a format string
AlsmPieceCount	Returns the number of pieces in a SuperString
AlsmPieceItemNo	Returns the item number to a corresponding piece
AlsmPieceSize	Returns the number of items in a piece
AlsmItemCount	Returns the total item count in a format string
AlsmItemType	Returns the data type symbol for an item

For more detailed information on SMS technology, please refer to the Functions section of the manual or online help.

List of Known Windows Problems Affecting App-Link iMS

The following is a list of Windows problems that we are aware of that affect the operation of the App-Link iMS controls.

- GPF under Windows 95 running multiple 16-bit applications

When running multiple 16-bit applications concurrently, shutting down the applications in a certain order may cause Windows 95 to crash. This problem is caused by a bug in the Microsoft OLE layer and can be resolved by applying their OLE32 Update. The file can be downloaded at: <http://www.microsoft.com/windows/software/oleupd.htm>. If you have not done so already, we also recommend that you download and install the latest Microsoft Windows 95 Service Packs.

- Windows NT 3.51 hangs unloading 16-bit applications

When running 16-bit applications concurrently, unloading 16-bit applications may cause Windows NT to hang. The problem is caused by a bug in the Windows NT WOW layer. To resolve the problem you must install Windows NT 3.51 Service Pack 4.

The file can be downloaded at:

<http://www.microsoft.com/bussys/winnt/winnt-public/fixes/usa/nt351/ussp4/i386>

In addition, you must configure any 16-bit applications to run in a separate WOW address space. This option is available when adding program items to a group window in the Program Manager. See the Windows NT 3.51 user documentation for more information on running 16-bit WOW applications in a separate address space.