



Mail²ZOS FTPSweeper

Concepts and Facilities

z/OS or OS/390

Release 1.0

February 12, 2006

Copyright and Ownership:

MailZZOS and FTPSweeper are proprietary products to be used only according to the terms and conditions of sale, lease or subscription. All software is the exclusive property of LaBayne and Associates, Fountain Valley, California. Every effort has been made to supply complete and accurate information.

Copyright 2003-2006, LaBayne and Associates, Inc. All Rights Reserved.

No part of this publication may be stored in a retrieval system, transmitted or reproduced in any way, including but not limited to photocopy, photograph, magnetic or other record, without the prior written agreement and written permission of the publisher.

Contents

<i>Chapter 1</i>	Mail ² ZOS	1
	Overview.....	1
	The Problem	1
	The Solution	1
	Flow of Data	3
	Features	4
	Mail2ZOS with JES2Mail	6
<i>Chapter 2</i>	FTPSweeper	7
	Overview.....	7
	The Problem	7
	The Solution	7
	Features	7
	Flow of Data	9
	Summary	10

Mail²ZOS

Overview

Mail2ZOS provides an MVS, OS/390, or z/OS system with the capability to receive (or retrieve) email messages with their attached data files into the mainframe, store the data into host files, submit jobs to process the data, and finally; send email notices to keep appropriate individuals informed of the operations.

The Problem

How to empower users without mainframe skills to get externally sourced data into the mainframe, initiate processing, and get results; without having to impose on mainframe operational staff resources at every step.

Today a variety of mechanisms are used to send data files to the mainframe. Some methods are direct, but run into security issues, and some indirect, but only move the data to a holding area where someone else has to take over. So, the first challenge is how to get the data into host files and available for mainframe operations.

The next step usually involves operations staff or programmers using TSO to modify JCL and submit jobs to process the newly updated files. Finally, operations staff must then wait for completion of activities, take appropriate action and notify responsible groups. It's a costly process that eats up both human resources and elapsed time.

The challenge is how to initiate the entire mainframe processing cycle – from data transfer, data decrypting, updating of JCL parameters, and sending/receiving of processing notifications – from outside the mainframe using simple, everyday technologies and methods – and in the process leverage operational human resources and reduce operational costs.

The Solution

The most convenient way to move data around is email. The simplicity to “send and forget”, the elegance of the store and forward transport mechanism, and the universality of support on nearly every known computing platform makes email the ideal way to transport data. **Mail2ZOS** allows data files to be emailed to the mainframe.

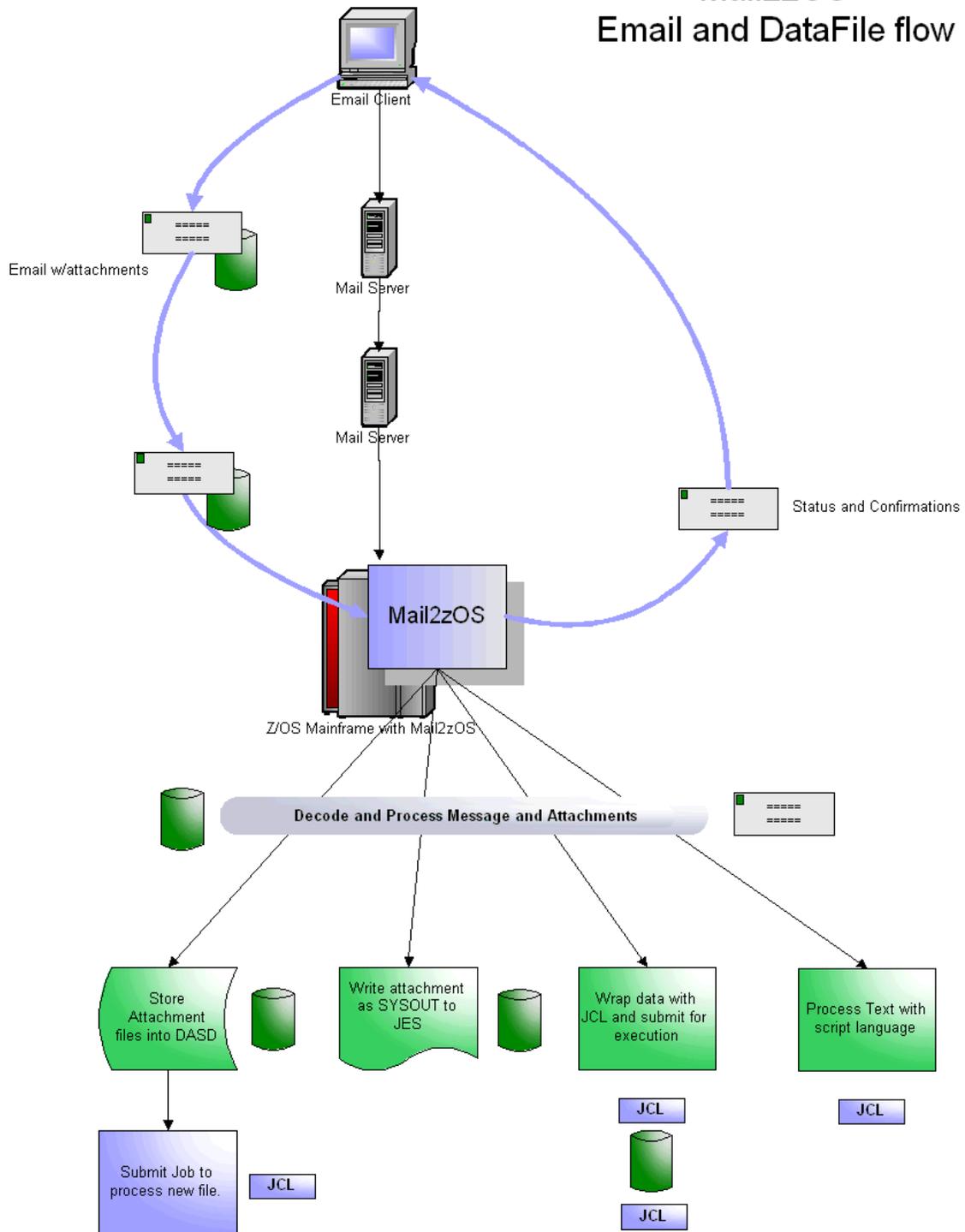
The principal functions of **Mail2ZOS** are to:

- ✓ Receive email “as an SMTP (post office) server”
- ✓ “Pick up” email from any number of POP3 servers
- ✓ Decode attached files and store them into MVS datasets
- ✓ Perform actions on the body or attachments:
 - Save to a host file and submit a job to process
 - Write the file to JES as a SYSOUT (printout)

- "Wrap" JCL around the data and submit for execution
 - Dynamically compose a job stream (JCL) and submit it for execution
- ✓ Send email notifications to keep appropriate individuals informed
- ✓ Use the script language for more extensive processing of the message body (cover page) and attachments

Flow of Data

Mail2zOS Email and DataFile flow



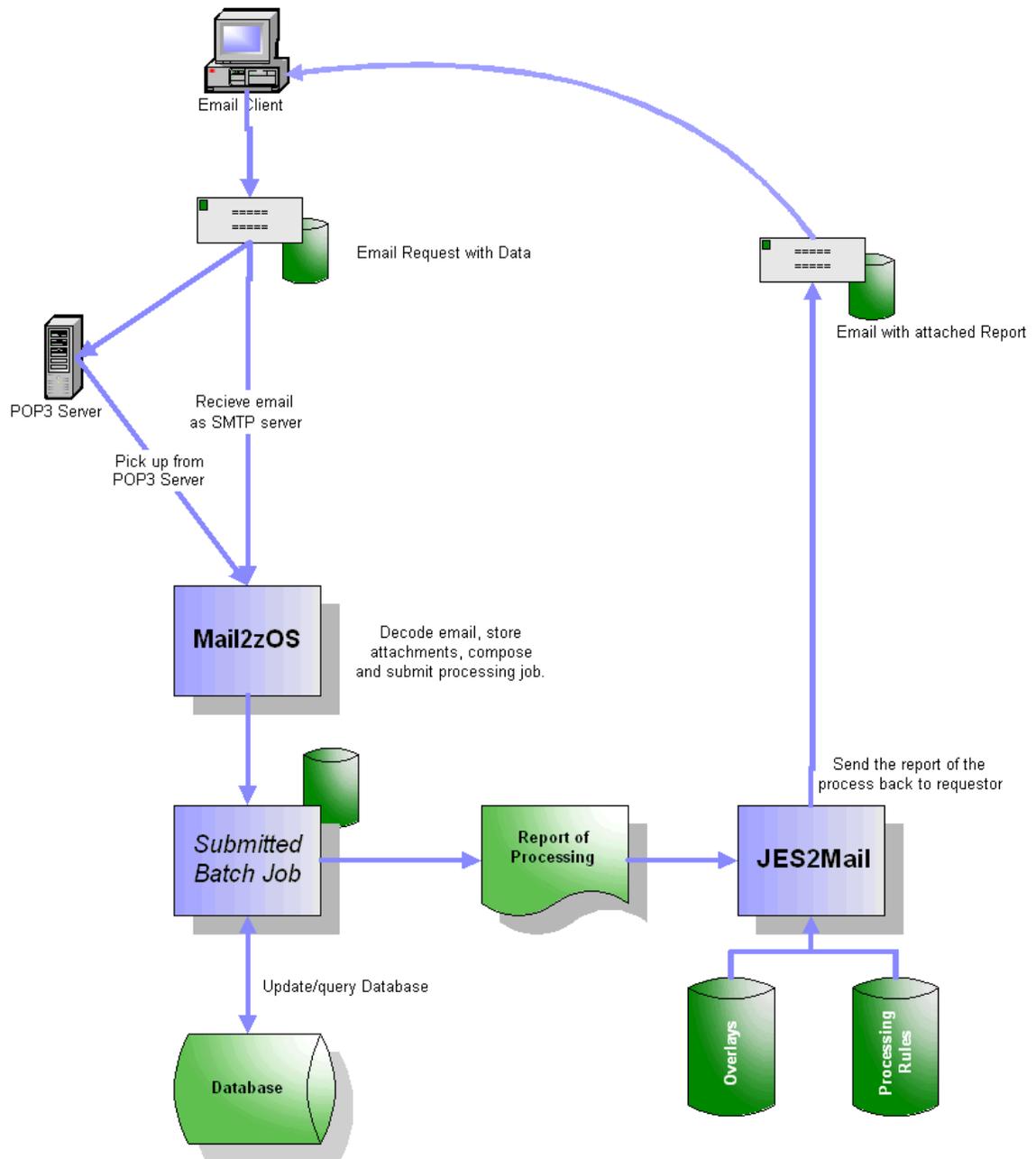
Features

- Run as Batch step or Started Task
- Listen for and receive email as an SMTP server
- “Pick up” email from any number of POP3 servers
- Safely store the raw email on the host HFS for processing
- Parse the email, separating the message body from attachments
- Allow specific processing for individual attachments
- Handle Binary Attachments:
 - Write as “blob” for “pure” binary data streams
 - Parse out fixed length records
 - Support RDW for variable length records
- For message “body” and text attachments
 - Translate from ASCII to EBCDIC
 - Save to fixed length (with padding) or variable length files
 - “De-edit” comma (or tab) delimited files into fixed fields
 - Sanitize unwanted characters, such as removing control chars
 - Transform data to upper or lower case
- Create unique filenames for the stored MVS filename
- Symbolic variable substitution
 - Insert the new “unique” filename into the JCL of the job being submitted
 - Conditional logic to dynamically alter the JCL being submitted
 - Symbolic variables include system related variables and most variables associated with the email, such as From, Subject, attachment filename, etc.
- Write files to JES as a SYSOUT
 - Most of the parameters from the JES OUTPUT statement are supported
- “Wrap” JCL around a data file and submit it for execution
- Dynamically compose a job stream (JCL) and either submit it directly, or store it in a data file for later execution (by a job scheduling system)
- Send email notices of messages processed and actions performed
- Support primary and backup post offices for sending notices
- TSO Notify for key events
- Five levels of tracing for very detailed or very summarized logging of activity
- “Pass” the text attachments or message body directly to a script for more extensive operations
- Script Language Support
 - High level script language
 - Can process the message body (cover page) or text attachment file directly
 - Perform table lookups to externalize processing data, i.e. passwords or the like
 - Text processing functions to:
 - Search and extract data lines for contents
 - Parse data elements from text line
 - Integer arithmetic

- String functions help manipulate data values
- Can dynamically create a job stream, more than just submitting a “static” job stream
- Job streams can be submitted to internal reader or created in a new unique file for later submission and execution.
- Dynamically create email notices
- Can invoke DLL routines in C, COBOL or Assembler to do additional operations. For example, after creating a job in a file, a DLL routine could link to a job scheduler to schedule the newly created job for execution.

Mail2ZOS with JES2Mail

While **Mail2ZOS** is all about “receiving and processing” email, JES2Mail is all about sending reports via email. Here is a flow of an application that receives requests via email to update a database, which in turn generates output that is sent back to the original requestor.



FTPSweeper

Overview

The purpose of **FTPSweeper** is to connect to remote FTP servers and “sweep up” data files, and when they are safely stored on the host, compose and submit JCL to process them.

The Problem

A z/OS or OS/390 mainframe is great at processing batches of data, but the problem is how to get a data file to the host, store it safely, submit a job to process it, and get notification whether everything is okay, all in an automated unattended secure operation.

There are all sorts of tools and mechanisms for transaction oriented data, but getting a “batch of data” to the host for processing seems to be a forgotten requirement, leading to some unsupported or unmanageable “Rube Goldberg” concoctions.

Often the off-host data is available to an FTP server or can be made so by simply copying the file to a network directory. The host may need to check lots of FTP servers for data files of interest, and only after the files have been safely uploaded and stored on the host, should the original file(s) be deleted (or renamed or kept “as is”).

Such a process should “run itself”. The sweep operation should occur on a regular basis, files should be uploaded, jobs submitted, notification emailed, all without manual intervention on either the originating or host side.

The Solution

The principal functions of **FTPSweeper** (or **HFSSweeper** mode) are to:

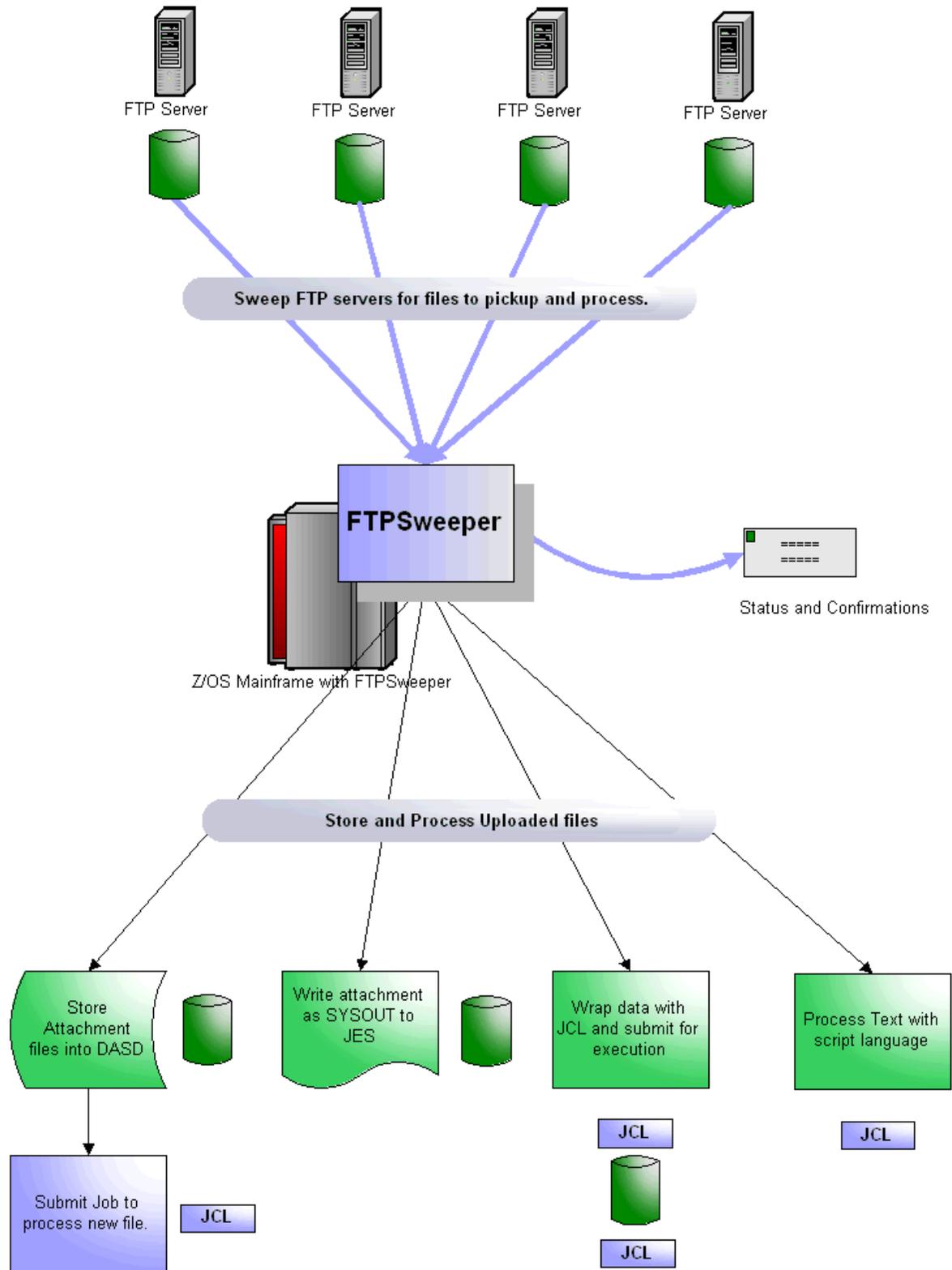
- ✓ Sweep data files from any number of FTP servers to the host
- ✓ Store them safely in MVS or HFS datasets
- ✓ Dynamically compose job streams (JCL) and submit them for execution
- ✓ Send email notifications to keep appropriate individuals informed
- ✓ Use the script language to extend the capabilities for processing files

Features

- Runs as Batch step or Started Task
- ‘Sweep’ any number of FTP sites or HFS directories
- Upload binary files:
 - Treat as “blob” (stream of binary byte data)
 - Parse into fixed length records
 - Support RDW for variable length files
- Upload text files:

- Translate from ASCII to EBCDIC
- Support for both built-in standard and IBM ICONV translation
- Save to fixed length (with padding) or variable length files
- “De-edit” comma (or tab) delimited files into ‘fixed’ fields
- Sanitize unwanted characters, such as removing control characters
- Transform data to upper or lower case
- Save files to MVS, including GDG, or HFS datasets on host
- Dynamically create unique filenames
- Symbolic substitution for a variety of variables
- Index (a number) can be incremented to create unique filenames
 - i.e.: FILE## => FILE01, FILE02, etc.
 - Write files to JES as SYSOUT
- Most of the parameters from the JES OUTPUT statement are supported
- “Wrap” JCL around data and submit to JES for execution
- Send email notices of files “picked up” and processed
- Send “audit” email notice of processing and errors
- Extensive logging and diagnostics with five levels of detail
- Support primary and backup post offices for sending notices
- Dynamic JCL creation and submission:
 - Symbolic variable substitution when writing JCL or email notices, etc. (i.e. Insert the name of the newly saved file into the JCL).
 - “Logic” statements to control the creation of the job stream
- TSO Notify for success or failure messages
- “Pass” the text data directly to a script for more extensive operations
- Script Language Support:
 - High level script language
 - Allows for more control over the FTP sign-on process and file selection process
 - Can process the data file directly
 - Text processing functions to:
 - Search and extract data lines for contents
 - Parse data elements from text line
 - Integer arithmetic
 - String functions help manipulate data values
 - Perform table lookups to externalize processing data, i.e. passwords or the like
 - Can dynamically create a job stream, more than just submitting a “static” job stream
 - Job streams can be submitted to the internal reader or created in a new unique file for later submission and execution.
 - Dynamically create email notices
 - Can invoke DLL routines in C, COBOL or Assembler to do additional operations. For example, after creating a job in a file, a DLL routine could link to a job scheduler to schedule the newly created job for execution

Flow of Data



Summary

Legacy mainframe applications continue to perform mission-critical processing while the overall number and knowledge level of mainframe users and technical human resources continue to diminish through retirement and overseas outsourcing.

Mail2ZOS and **FTPSweeper** are innovations which use everyday technologies like email and FTP for data updating and application processing, further leveraging an organization's shrinking mainframe knowledge and skill base.

Please visit www.casisoft.com/m2zhome.html for more information, or contact us by email at info@casisoft.com or by telephone at 714-378-1109.

CASI Software

10231 Slater Avenue, Suite 117 • Fountain Valley, CA 92708 • 714-378-1109
www.casisoft.com