Implementing SAS IT Charge Manager™ at the State of Wisconsin
Bruce McClinton, Hudson Williams Inc., Maurertown, Virginia

ABSTRACT

The State of Wisconsin Department of Administration supports the diverse needs of state agencies for Information Technology services. The rapid growth of new programs such as the nationally recognized Wisconsin Works (W2) has put added demands on the resources of the department.

The cost of these services must be recovered in an equitable fashion from client agencies. The department evaluated several accounting and chargeback tools in a search for one that would combine flexibility, and ease of administration. And position them to deliver chargeback information to the desktops of their customers.

This paper describes our experiences in replacing CA-MICS in the installation of the new SAS IT Charge Manager product. We'll examine the features that influenced the State's choice. And how the State's reporting requirements resulted in some creative product tailoring.

INTRODUCTION

The State of Wisconsin Department of Administration (DOA) is legally bound to charge back to the state agencies the costs associated with the use of its shared systems. Chargeback of mainframe systems has been in use at the State for over 15 years.

As with many modern data centers, the DOA is the result of consolidation. The department was created to replace three independent agency centers. Each had evolved their own set of standards and software. When consolidations of chargeback systems occurred they were often rushed. There was little time to review the differences between systems. This adversely impacted the overall architecture, system performance, and maintainability of the combined system. When the DOA determined the time was right to replace their current system, they took the opportunity to assess chargeback from the top down.

First the DOA reviewed the chargeback methodology that was in place. Chargeback methodologies are unique to each company and include:

- Selection of billing units
- Identification of users
- Rate setting procedures
- Cost pool development
- Allocation of indirect expenses
- Invoicing and information delivery

In particular the DOA was concerned with the manner in which indirect expense, also referred to as "non-billables", were distributed. The consolidation of data centers, which created the DOA, had caused inconsistent distribution of this expense.

Another chargeback methodology issue that was addressed was invoicing and information delivery. DOA staff spends significant time assisting customers with their bills. Despite this, the customers sometimes do not have the information they need to understand and effectively manage their IT costs. In response the DOA is examining the option of delivering chargeback information to the customers desktop.

WHY SAS IT CHARGE MANAGER?
Every chargeback implementation is a unique entity, and the software that supports it must adapt to that character.

The State selected SAS IT Service Vision™, and the companion product SAS IT Charge Manager. These two products provide several advantages over competitive products:

- Vastly more flexible than 'turn-key' solutions
- Substantially reduced maintenance when compared to MICS
- An open data architecture making it easier to provide customer access to data
- A common data repository, interface, and toolset for chargeback, performance management and capacity planning.

HUDSON WILLIAMS
From the beginning of the selection process DOA realized that they would need a partner with broad experience in Accounting and Chargeback. Also, they needed someone who was familiar with both CA-MICS and the SAS products.

Hudson Williams specializes in systems management of the disparate environments common in today's IT organizations. Two years ago we perceived the need for a chargeback product to complement the power of SAS IT Service Vision and set out to create it. That product is now known as SAS IT Charge Manager. It was developed from our staff's experience consulting on accounting and chargeback implementations at many Fortune 1000 companies.

CONVERSION
The process of migrating from CA-MICS to SAS IT Charge Manager began with a detailed review of the existing chargeback system by our staff. More than a year passed before all the players and elements were in place to support the conversion. In the following sections we will review:

- The issues which influenced the design
- Product tailoring and customization required to meet the design objectives
- Results of parallel testing
- Operational experiences with the new system
- Future plans

ISSUES

TWO VIEWS
Proponents of products such as SAS IT Service Vision and CA-MICS emphasize the advantages of a common repository for all systems management data. The cost of parsing the complex raw data into the database has always been the primary argument in favor of this approach. Management at DOA argues that other factors out-weigh this cost.

The functions commonly supported by a performance measurement repository commonly include:

- Chargeback and Accounting
- Capacity Planning
- Performance Management
- Availability Tracking

There is considerable overlap in terms of data source used by these functions. However, each demands different elements, summarization, and periods of retention. An ideal solution might be to have a single data parsing process that feeds

multiple views of the data. This is possible using the ITSV architecture. However, other considerations ruled out that approach.

Like many organizations the DOA separates responsibility for performance and chargeback into different organizations. Changes made by one organization to the repository can have critical impact on other functions. In such situation the chargeback function would be at greatest risk. Because of this issue and the differences in data requirements the DOA has chosen to implement separate databases and tailoring libraries for each function.

EXTRACTS AND INVOICES
CA-MICS and SAS IT Charge Manager share a common approach to charge generation. Rates are applied, and charges generated each day as raw data is processed into the system. The SAS table containing parsed raw data is known in MICS as a Data Integration Product file. The accounting product produces its own file containing the identifying information, quantity, rate, and charge. For the sake of brevity we shall refer to these two tables respectively as the 'Charged Table', and the 'Accounting Table'.

DOA customers are provided with monthly charge information in two formats. An invoice is produced which summarizes charges by agency. To support the invoice a flat file, known as an extract file is provided which breaks down the charge by job, user and transaction. The extract also includes non-billing related elements to assist the analyst in understanding the source of the charge.

In the old system the extract files were created directly from the charged table. These charged tables contained all of the identifying information that the user might require. Also, the CA-MICS Accounting and Chargeback Product added an element to each charged table that contained the total charges associated with a single observation.

We considered using the accounting tables in SAS IT Charge Manager to support the extract function. This would have resulted in some space savings. However, it would have required changes to the accounting table definitions that would have been difficult to maintain.. We chose instead to replicate the approach taken in CA-MICS.

We decided to add the necessary identifying and charge elements into the charged tables. Since the definition of these tables was under our control there was little chance of conflict with future maintenance.

TIME PRESSURES
As often happens, events conspired to move us from a careful Needs Analysis into a rapid implementation. The implementation had to be completed before the start of the State's fiscal year. In order to meet this deadline we were forced to limit the process re-engineering phase. The emphasis shifted to identifying and moving active code from CA-MICS to SAS IT Service Vision and SAS IT Charge Manager.

THE DESIGN

ISOLATING THE SITE
We needed to decide at which level within the product we would isolate the chargeback and performance databases from each other. That is, what portion of the product implementation would be shared, and what portion would be unique to each function.

A SAS IT Service Vision MVS installation consists of the following elements:

- The product installation libraries
- The MXG installation libraries
- A SITELIB containing global product customizations
- A library containing local source customizations and overrides to MXG
- One or more Performance Databases (PDBs)

Given the expected level of customization, and the need to maximize stability of the chargeback system we elected to share only the two sets of product install libraries with the non-chargeback users of the product.

By using a separate SITELIB we had created the equivalent of separate, function specific CA-MICS complexes. But, without the cost of a complete set of installation libraries.

PERFORMANCE DATABASES
A PDB is associated with a set of summarization libraries and update processes. This is the equivalent of a CA-MICS Database Unit.

In deciding how many PDBs to create we considered:

- The location of the raw data and the time when it would be available
- How best to balance the volume and processing time required for each data source
- How to best fit the processing within the available nightly batch window
- The impact of occasional PDB update restarts during prime shift

We selected the following structure:

PDB	Data Source	Comments
CICS	CMF 110	4-5 million trans/day. Segregated SMF file.
DB2	DB2 101 & 102	2+ million trans/day. Also segregated.
IMS	IMS Log	2-3 million trans/day. Non-SMF collection.
SMF	SMF 6, 26, 30	Moderate batch and TSO
	Nomad	Low volume database activity. In same SMF dataset as the above.
	IDMS PMDC	
STORAGE	DCOLLECT	Combined collection run and PDB update.

EXTERNAL CHARGES
Typically chargeback systems include data from a variety of sources that are specific to chargeback, and not required for other functions. In CA-MICS these are accommodated by the Accounting and Chargeback Product's External File Interface. At the time this paper was written no equivalent facility in SAS IT Charge Manager. However, the Generic Data Collector feature of SAS IT Service Vision provides a simple mechanism for adding this type of data.

Processing external charges is often complicated by the fact that the data arrives at different times of the month. Fortunately, SAS IT Service Vision permits updating of individual tables, rather than all files at once as required by CA-MICS. That made it practical to create a single PDB to accept data from a variety of low volume and ad-hoc sources:

Data Source	Frequency	Comments
Advantis	Monthly	Passthrough charges
Cntl-T	Daily	Tape occupancy
EMail	Monthly	Per mailbox charge from MS-Access extract
EOS	Weekly	Report Distribution System
File Handler	Monthly	C/S Resource Accounting
Optical DASD	Montly	Snapshot of storage occupancy

CHARGE ITEMS
Since this was a conversion the list of charge items was already defined. We were left to identify the appropriate tables and select or create the elements from which to charge. Figure 1 shows the resulting list.

Figure 1 – Charge Items

PDB	Table	Charge Item #	Item Description	Units
CICS	XCITRAN	500015	CICS Transactions	TCB CPU Secs
DB2	XDBACCT	500014	DB2 – CICS Transactions	TCB CPU Secs
IMS	XIMTRAN	500016	IMS Transactions	TCB CPU Secs
SMF	UNOMAD	500032	Nomad Transactions	TCB CPU Secs
	UPMTASK	500022	IDMS Batch Prime Trans	TCB CPU Secs
		500023	IDMS Batch Non-Prime Trans	TCB CPU Secs
		500024	IDMS Batch Weekend/Holiday	TCB CPU Secs
		500025	IDMS Batch Premium Trans	TCB CPU Secs
	XJOBS	500036	Batch Prime	TCB CPU Secs
		500037	Batch Non-Prime	TCB CPU Secs
		500038	Batch Premium	TCB CPU Secs
		500039	Batch Weekend/Holiday	TCB CPU Secs
		500040	TSO	TCB CPU Secs
	XPRINT	500005	Impact Print	Thousand Lines
		500006	AFP/XEROX Print	Pages
		500007	Fiche Output	Frames
		500008	Class K Fiche	Frames
Storage	XDCBKUP	500009	Backup Datasets	Allocated Mbytes * Days
	XDCDSET	500012	SMS Datasets (never archived)	Allocated Mbytes * Days
		500013	SMS Datasets (eventually archived)	Allocated Mbytes * Days
	XDCMIGS	500011	HSM ML2 Datasets	Allocated Mbytes * Days
		500021	HSM ML1 Datasets	Allocated Mbytes * Days
External	UADVTNS	500026	Advantis Charges	Dollars
	UCNTLT	500027	Tape Occupancy	Allocated Mbytes * Days
	UEMAIL	500028	E-Mail Service	Clients * Months
	UEOSRPT	500029	Electronic Report Distribution	Thousand Lines
	UFHTRNA	500034	File Handler Alpha	TCB CPU Secs
	UFHTRND	500035	File Handler Database	TCB CPU Secs
	UOPTDAS	500033	Optical DASD Usage	Mbytes * Months

THE IMPLEMENTATION

The similarity between CA-MICS and the SAS IT Service Vision and SAS IT Charge Manager products made the process of migration relatively easy. This similarity extends beyond the fact that they are both based primarily on the SAS System. They also have similar application architectures. In many cases once modules were reviewed for applicability they could be converted by simply changing variable references.

The most difficult part of this process was reviewing the existing modules to identify obsolete or dead code. Fortunately one of our team members (Will Thompson) had long experience with the existing system and was able to identify most of the problem areas.

The following sections detail the steps required to tailor and customize SAS IT Service Vision and SAS IT Charge Manager to meet the design objectives.

CUSTOMER IDENTIFICATION

Account code assignment is performed at two levels within CA-MICS. The Account Code Definition and Derivation modules create the Account elements in the Data Integration Product (non-chargeback) files. The Cost Center Definition and Derivation modules define the account codes used in the Accounting and Chargeback product. These fields do not need to be the same. Although the Cost Center fields are frequently a subset or superset of the Account fields.

Account code elements exist only in a few tables in SAS IT Service Vision. They are not ordinarily required because of the ease with which the Class (summarization) variables can be

modified. The account codes instead are assigned in SAS IT Charge Manager. These are called Customer Name fields.

The Customer Identification routine is a large module, which is invoked for all charged tables. The example below is taken from the sample routine provided with the product:

```
%IF &CPTABLE EQ %UPCASE(XJOBS) %THEN %DO;
  E_CUST1 = PUT(SUBSTR(JOB,1,2),$CMPFMT.);
  E_CUST2 = PUT(SUBSTR(JOB,1,2),$DIVFMT.);
  E_AUDID = 'JOB='      ||JOB      ||' '||
            'JESNUM='   ||JESNMR  ;
%END;
%ELSE %IF &CPTABLE EQ %UPCASE(XDASD) %THEN %DO;
  E_CUST1 = PUT(SUBSTR(DSNHLQ,1,2),$CMPFMT.);
  E_CUST2 = PUT(SUBSTR(DSNHLQ,1,2),$DIVFMT.);
  E_AUDID = 'HLQ='      ||DSNHLQ  ||' '||
            'VOLSER='   ||VOLSER  ;
%END;
```

In order to simplify maintenance we elected to move the assignment logic into individual members of an external PDS based on the tables supported. By including these smaller modules into the Customer Identification routine we made them easier to manage and access.

While account fields are normally not required in the charged tables, we needed them in order to support creation of the Extract files. To accomplish this we 'back filled' the elements during the table merge process in %CMPROCES. At this point the DATA step is creating both the charged table (e.g., XJOBS), and the corresponding SAS IT Charge Manager table (e.g., EJOBS). So the elements of each are available to the other.

First we created the new elements as Derived Variables in the charged table. Shown below is an example of the data dictionary utility (%CPDDUTL) statements used:

```
CREATE DERIVED NAME= E0CUST1
  FIXED =NO
  LABEL='Agency/Customer Number'
  DESCRIPTION='ITCM Agency/Customer Number'
  KEPT=YES INTERPRET=STRING TYPE=CHARACTER
  LENGTH=8 FORMAT=.
  SUBJECT='ITCM'
    SOURCE={
    }
;
CREATE DERIVED NAME= E0TCHRG
  FIXED =NO
  LABEL='Total Charges'
  DESCRIPTION='TOTAL CHARGES ASSESED TO THIS
OBSERVATION BY ITCM'
  KEPT=YES INTERPRET=SUM TYPE=NUMERIC
  LENGTH=8 FORMAT=DOLLAR12.4
  SUBJECT='ITCM'
    SOURCE={
    }
  DAY =( SUM )
  MONTH =( SUM )
;
```

The next step was to add the code necessary to assign the value of the elements. Normally a Derived element is populated by code in the 'SOURCE=' keyword of the CREATE statement. However, this occurs at the wrong point in the logic for our application. Instead, we modified the PROC205 exit point as shown below:

```
IF E_NEW=1 THEN DO;
   %E_PROC(DATECHK=&DATECHK);
```

```
   e0cust1 = e_cust1 ;
   e0cust2 = e_cust2 ;
   e0tchrg = e_totchg ;
   e0invsd = e_invsd ;

   END;
 E_NEW=0;
```

The *italicized* statements were added. The other statements are part of the standard SAS IT Charge Manager exit routine. We were concerned about modifying a supplied module. But, this module is static, and probably will only change occasionally. Hopefully future versions of SAS IT Charge Manager will eliminate this issue by isolating product code from the user exits.

The elements added include:

- E0CUST1 – first level of the customer name hierarchy, the Agency Code
- E0CUST2 – second level, the Account or Project Number
- E0TCHRG – total charge for this observation (similar to the fffCOST element in CA-MICS)
- E0INVSD – the invoice start date. This allows us to correlate charges from this file back to the invoice.

CHARGE QUANTITIES
We discovered that it was important to have the quantity charged available in both the charged, and charges tables. We made it a practice create the quantity field as a Derived Variable in the charged table. This was particularly important for CPU and print charges.

Print charging is always challenging for MVS systems. In this case we needed to accommodate charging for three different types of devices from the XPRINT table: impact printers, laser prints, and fiche machines. Each has a different charging unit. We created a single derived element whose logic encompassed them all.

```
IF PUT(OUDEVCE,$PRNTTYP.) IN ('L' 'X')
THEN DO;
   IF SUBSYS6='PSF' THEN
    LINEPGE=PAGECNT;  /* PSF */
   ELSE LINEPGE=CEIL(PAGECNT
    /(PUT(OUDEVCE||FORM,$PAGEDEF.)));  /*
XEROX */
END;
ELSE IF PUT(OUDEVCE,$PRNTTYP.) IN
 ('I' 'Z') THEN
 LINEPGE=PRNTLNE/1000;  /* IMPACT */
ELSE IF PUT(OUDEVCE,$PRNTTYP.)='F'
 AND OUCLASS IN ('M') THEN
 LINEPGE=INT((PRNTLNE/11024)+.999); /*
FICHE */
ELSE IF PUT(OUDEVCE,$PRNTTYP.)='F'
 AND OUCLASS IN ('K') THEN
 LINEPGE=INT((PRNTLNE/14310)+.999);
ELSE LINEPGE=0;
```

All CPU charges are based on the TCB CPU time of the job or transaction normalized to a base processor speed. This is done in a common macro. The macro, shown below, is invoked to create the NRMLTCB derived element in each charged table:

```
%MACRO CPUFAC(LPAR,ENDTS,FACTOR);
IF &LPAR IN ('DEV1' 'PRD2' 'ITS1') THEN
DO;
   IF &ENDTS>'23MAR97:0:0:0'DT THEN
      &FACTOR = 2.0;
   ELSE &FACTOR = 1.0;
```

```
    END;
ELSE
 IF &LPAR IN ('PRD1' 'CMC1') THEN DO;
    IF &ENDTS>'03JAN99:04:0:0'DT THEN
        &FACTOR = 2.0;
    ELSE &FACTOR = 0.93;
    END;
ELSE
    &FACTOR=0;
%MEND CPUFAC;
```

SYSPLEX CONSIDERATIONS

The IBM Parallell Sysplex™ facility used in combination with IBM SmartBatch™ has implications for chargeback. Used together they can send the steps of a single job to whichever system images have available resources. If these images happen to be running on different physical machines then the CPU normalization factor must be varied by step. This can accommodated by charging from the XSTEPS table rather than XJOBS. Unfortunately, this was not an option for us.

The charge extract file had to be summarized by job. Rather than reproduce the MXG step summarization logic we chose to modify it. We added the normalized element to the step file, and summarized it to the job level.

First we added the following logic to the MXG EXTY30U4 module to create the normalized TCB element:

```
/* NORMALIZE CPU TIME */
  %CPUFAC(SYSTEM,TERMTIME,CPUFAC) ;
  NRMLTCB = CPUTCBTM*CPUFAC ;
```

The element was added to two macros in the IMACPDB member. The _PDB30_4 contains the list of variables retained in the PDB. It also controls which variables are available to the JOBS file summarization logic. The _SUMSTP macro causes it to be summed across step observations when building the JOBS MXG file which becomes the XJOBS SAS IT Service Vision table.

SHIFT CHARGING

IT Charge Manager does support Shift Charging. However, all shifts are assigned to a single charge item. This was not consistent with the existing charge item list. Instead we used the charge derivation routine to limit individual charge items to a specific shift. For example:

```
/* COMPUTATION ROUTINE Job Prime CPU Consumed */
if tyetask eq 'JOB'
 and joclass not in ('P','Q')
 and shift eq '1'
 then e_cqty = nrmltcb;
```

KEEP LIST TAILORING

One of the objectives of this product conversion was to reduce the size of the chargeback database. We also wished to reduce the size of the work files and update resources. This was particularly important for high volume data such as CICS, DB2, and IMS.

Setting an element to KEPT=NO only effects the size of the PDB libraries. It has no impact on the work files produced by MXG. To reduce the work files you must modify the underlying MXG source. Initially we did this by creating a utility which produced a DROP= which we added to the MXG IMACxxxx members.

Subsequently we discovered a note in SAS CPE Newsletter 5 that describes a technique to automatically link the MXG keep list to the data dictionary. The newsletter is still available on the SAS Institute web site:

http://www.sas.com/software/it_vision/document/newsletters /CPNEWS/5.REVKEEP.html#I5.A53.M

DATETIME MODIFICATION

One significant difference between CA-MICS and SAS IT Service Vision did impact us. The DATETIME field is the priamry timestamp in SAS IT Service Vision tables. For most MVS tables this field contains the start time of the interval or event. DATETIME is also used to determine the value of SHIFT. CA-MICS uses the ENDTS variable to determine the ZONE (shift).

This resulted in some variation in cost recovery for batch activity as jobs moved from one shift to another. To resolve this we modified the definition of DATETIME in the PROCxxx exit routine prior to the derivation of the SHIFT variable.

```
%macro proc040 ;
 %***
 %* PROC040 Exit Code
 %*** ;

 %IF &CPTABLE EQ XJOBS %THEN %DO ;
   /* change for xjobs table */
 if jtmtime ne . /* 30-5 job term  */
    then datetime = jtmtime ;
 else if jedtime ne . /* 26 job term    */
  then datetime = jedtime ;
 else if temtime ne . /* last 30-4 term */
  then datetime = temtime ;
 else if jprtime ne . /* purge time     */
  then datetime = jprtime ;
 shift = '' ;
 %end ;
 %mend proc040 ;
 %proc040
```

PRODUCTION

We began installing and customizing the new system at the end of April 1998. By the first week of June we were ready to begin parallel testing the systems.

During June we did daily comparisons of charges from the two systems. A few problems were discovered, particularly in print charging. But, by the end of the month we were confident that the new system was ready for production.

Just prior to July 1st we purged the database tables. Charges began accumulating from the start of the state's new fiscal year. However, to be safe we continued parallel updates of CA-MICS and SAS IT Service Vision.

VALIDATION

Formal validation of SAS IT Charge Manager came during the first week of August. We had of course been comparing the charges generated by each system every day. Given the differences in product architectures we anticipated a month-end charge variance of up to one percent. But we were very pleased, and quite surprised to discover a variation of just a few cents.

EXPERIENCES

In the months since the production implementation was completed Hudson Williams has continue to provide on-call assistance to the DOA. The experience has been positive for both organizations. The DOA has benefited from having experienced technicians available to help while their staff gains experience. We have had an opportunity to experience the pitfalls of day-to-day operations of the product. There have been a number of lessons learned along the way.

STORAGE

During the early month of production operation we encountered frequent update failures most commonly due to insufficient space in PDB libraries. In future we hope to have utilities available which will permit better estimation of the storage requirements of a fully populated system.

RECOVERY

A continuing problem has been backup and recovery. The DUPCHECK facility of SAS IT Service Vision (similar to CA-MICS Checkpoint) should in theory permit safe step level restart. However, we have not had an opportunity to do rigorous testing to verify this.

Therefore our standard procedure has been to restore the entire PDB prior to a restart. Unfortunately, many of the database libraries are multi-volume. These cannot be successfully recovered with standard IBM DFSMSdss™ product backups. We are now converting over to SAS PROC COPY for backups. These are much slower. We are continuing to search for a reliable disk image backup utility.

Anyone who anticipates the need for multi-volume library allocation should carefully read the available material on the subject in the SAS Companion for the MVS Environment, Version 6, Second Edition, and applicable usage notes. We also suggest you consult your storage management staff.

ADJUSTMENTS

Recently we experienced a small error in charging when it was discovered that the SAS IT Service Manager holiday definitions had not been updated for 1999. As a result batch CPU charges for January 1st were incorrectly assigned.

We were able to extract, and in one case recreate these charges from the original data. We applied the calculated credits for these charges back into the system by creating a generic collector as a pass-through mechanism for the charges. The variable containing the credit was assigned as a charge item, and a rate of one dollar was applied.

The standard facility for debits and credits works adequately for low volume adjustments. But, until such time as a charge import facility is added to SAS IT Charge Manager this type of generic collector table will be a standard feature of all of our future installations.

RESOURCES

One of the objectives of this conversion was to reduce the cost associated with collecting charges. This has been achieved by providing views of the monitor data that are more closely tailored to the requirements. The saving in storage has been significant. We also found a modest reduction in run times and processor time consumed by the update jobs.

CLIENTS SOFTWARE

Both SAS IT Service Vision and SAS IT Charge Manager feature client software which rely on SAS/CONNECT®. The DOA staff felt that the level of security offered by version 6 of SAS/CONNECT with TCP/IP was too low to permit the widespread use of the client software. Future changes to SAS/CONNECT should resolve these issues. But we recommend that you review the use of SAS/CONNECT with your network staff.

FUTURE ENHANCEMENTS

Some of the early objectives of this conversion have been delayed by more pressing work. However, work has begun to re-engineer the maintenance of the customer identification tables. Still under consideration is a project to provide customers with better tools for understanding the charges and managing their costs.

CONCLUSION

As with many chargeback projects, a late start and short deadline left us feeling that we could have accomplished more. However, the basic features of the system were in place by the target date. The SAS IT Service Vision and SAS IT Charge Manager products met the primary objectives of the DOA staff. And after a few initial teething pains, the system is operating smoothly.

REFERENCES

SAS Institute Inc. SAS Companion for the MVS Environment, Version 6, Second Edition, Cary, NC: SAS Institute Inc.

ACKNOWLEDGMENTS

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Bruce R McClinton
Hudson Williams Inc.
7714 Back Road
Maurertown, VA 22644
Work Phone: 540-459-9364
Email: bruce.mcclinton@hudsonwilliams.com
Web: http://www.hudsonwilliams.com

Hudson Williams is a SAS Quality Partner, and a SAS IT Service Vision Solution Provider.