Release Notes for
MMS-EASE *Lite*

MMS-LITE-801-001

MMS-LITE-802-001

Revision 19

Systems Integration Specialists Company, Inc.
6605 19½ Mile Road, Sterling Heights, MI 48314-1408, U.S.A.
Tel: +1-586-254-0020, Fax: +1-586-254-0053
E-mail: support@sisconet.com, URL: http://www.sisconet.com

Printed in U.S.A.

03/2012

New and Changed Software Features

For information on changes and corrections for V5.3001, V5.10 and V5.06, please refer to the end of this release note.
For information on changes and corrections before V5.06, please examine Revision 15 of the Release Notes.

This release of MMS-EASE *Lite* (V6.0000) contains the following changes:

IEC 61850 Edition 2 Changes

- Changed the SCL parser to allow the new optional attribute “ix” in the FCDA element (*#2684*).
  The functions in **sclparse.c** and **sclproc.c** were modified to handle this. This works by including an AlternateAccess definition in the NVL to specify the array index for one or more of the NVL variables.

- GOOSE subscriptions are automatically created from “ExtRef” elements in the SCL file and the values of some LGOS attributes are set automatically (*#2818*) (MMS-LITE-802-001 version only).
  If GOOSE is supported, and ExtRef elements are found in the SCL file, an appropriate GOOSE subscription is automatically created at startup. When a GOOSE message is received, a “user” function is called to process it. A “user” function example is provided in **`callback_function`** in **iec_demo.c.** The user must modify or replace this function to process the GOOSE data appropriately.
  Note: If multiple ExtRef elements reference the same GOOSE Control Block, only one GOOSE subscription is created for all of them.

  If the **`setSrcCB`** attribute in LGOS is initialized in SCL and it references a valid GOOSE control block (like this):

```
<LN lnType="LGOS" lnClass="LGOS" inst="1">
    <DOI name="GoCBRef">
        <DAI name="setSrcCB">
            <Val>GOOSEPUBC1/LLN0$GO$ItlPositions</Val>
        </DAI>
    </DOI>
</LN>
```

  then you should see the following values in LGOS:

```
  LGOS$ST$NdsCom$stVal    =0
  LGOS$ST$St$stVal        =1 if receiving GOOSE, 0 if NOT receiving.
  LGOS$ST$SimSt$stVal     =0 (not configurable but user code could change this).
  LGOS$ST$LastStNum$stVal =last StNum value received in a GOOSE (0 if NOT receiving).
  LGOS$ST$ConfRevNum$stVal=confRev configured for this GOOSE control block.
```

  The **`confRev`** is configured in the GSEControl element in SCL, like this:

```
  <GSEControl name="ItlPositions" datSet="Positions" appID="Itl" confRev="5"/>
```

  If you change **`confRev`** in the SCL, you should see the new value in LGOS.

  If the **`setSrcCB`** attribute in LGOS is NOT initialized in SCL or it is incorrect, you should see the following values in LGOS:

```
  LGOS$ST$NdsCom$stVal    =1
  LGOS$ST$St$stVal        =0
  LGOS$ST$SimSt$stVal     =0
  LGOS$ST$LastStNum$stVal =0
  LGOS$ST$ConfRevNum$stVal=0
```

- Added code to automatically manage IEC 61850 Logs (*#2911*).
  If a "**LogControl**" element is configured in the SCL file, the code now automatically creates the Journal, and automatically generates Journal entries, much like the Reporting code. The default maximum number of Journal entries is 1000, but if you use the new processing function **`scl2_ld_create_all_scd`** , you can specify a new value. See documentation of **`scl2_ld_create_all_scd`**, especially the **`SCL_SERV_OPT`** argument in the *MMS-Lite Reference Manual* for more details.

- The SCL parser now automatically recognizes Edition 1 or Edition 2 SCL files and processes each appropriately (*#3067*).
  The parser checks for "version" and "revision" in the first line of the file.

  For Edition 2, the version must be "2007" and revision must be "A," but the parser will assume Edition 2 if these are present (no matter what the values). If they are not "2007" and "A," the parser just logs a warning.

  If version and revision are not present, the parser assumes Edition 1.

  The user can also override this logic by specifying the Edition in the "forceEdition" attribute of the **`SCL_OPTIONS`** structure passed to **`scl_parse_scd_all`**. The value of "forceEdition" should be 1 (for Edition 1) or 2 (for edition 2). If forceEdition = 0, the Edition detected in the SCL file is used.

- Added support for the new attributes "resvTms" and "owner" in Edition 2 RCBs (*#3068*).
  If an Edition 2 SCL file is detected, the SCL parser may automatically include the optional attributes "ResvTms" and "Owner" in RCBs.

  "ResvTms" is included (in each BRCB) if the SCL "ReportSettings" element contains resvTms="true".

  "Owner" is included (in each BRCB or URCB) if the SCL "ReportSettings" element contains owner="true".

  Caution: The "ReportSettings" element does not currently allow the "owner" attribute but Tissue #807 proposes to add this attribute. While Tissue #807 is being resolved, the user can choose whether "Owner" should be present. If the user sets the "includeOwner" flag in the **`SCL_OPTIONS`** structure passed to **`scl_parse_cid`** or **`scl_parse_scd_all`** (new functions), then "Owner" will be included in each BRCB or URCB.

- Support for SCL configuration of the Retransmission Curve for GOOSE (*#3073)*.
  The SCL parser uses the new "MinTime" and "MaxTime" elements in the "GSE" element to define the GOOSE retransmission curve. "MinTime" (in milliseconds) is used as the first retransmission time. Then the retransmission time is doubled with each retransmission until it reaches "MaxTime".

- Support for All New Logical Nodes and Common Data Classes using SCL (*#3075*).
  The SCL importer is now able to import and configure all new logical nodes for Edition 2 of IEC 61850-7-4, all new common data classes for edition 2 of IEC 61850-7-3 and all new service tracking common data classes for edition 2 of IEC 61850-7-2. Processing of the underlying functions will be handled by user code leaf functions. This code will be left to the discretion of the user to develop.

- A new sample SCL file is provided with all the Edition 1 CDCs and Logical Nodes updated for Edition 2. It also defines all the new Edition 2 CDCs, and the following new Logical Nodes (to demonstrate use of the new CDCs):

  LGOS
  LTRK (uses new "tracking" CDCs)
  QITR (uses new HST CDC)
  LTMS (uses new VSS and VSG CDCs)
  LTIM (uses new TSG CDC)

  Other new Logical Nodes must be added by the user, if needed. The user must provide the necessary code to read and write the data for these new CDCs and Logical Nodes (through "leaf" functions).

- Support for New Multicast Sampled Value Message and Control Blocks (*#3076*).
  MMS Lite now supports the new Edition 2 message format for the Sampled Values and the new control block parameters for the MVSCB per IEC 61850-7-2 and IEC 61850-9-2.

  The **scl_tpxs0** sample will encode Edition 2 Sampled Value messages if an Edition 2 SCL file is detected; otherwise, it encodes Edition 1.

  The **cositpxs0** sample receives and decodes Edition 2 Sampled Value messages. If the user wants to decode Edition 1 SampledValue messages, they must change the **`edition`** parameter passed to **`smpval_msg_decode`** (see **subnet.c**).

  Note: The **scl_tpxs0** sample demonstrates the data mapping specified in “Implementation Guideline for Digital Interface to Instrument Transformers using IEC 61850-9-2” from the UCA International Users Group (http://iec61850.ucaiug.org/Implementation%20Guidelines/DigIF_spec_9-2LE_R2-1_040707-CB.pdf).

- Added the option to get addressing information from the SCL file (*#3282*).
  An option was added to store **`DIB_ENTRY`** structures in a linked list and then add addresses extracted from the SCL file to the linked list. The addresses in SCL do not have anything like an AR Name, so the AR Name is generated automatically from the IED Name and AccessPoint Name, separated by "/". For example, if the SCL file contains an address entry, like this:

  <ConnectedAP iedName="E1Q1SB1" apName="S1">

  the AR Name generated is **E1Q1SB1/S1**.

  In the Server sample applications (**scl_srvr** and **scl_tpxs0**), the local address is the address for the IED and AccessPoint selected in **startup.cfg**. Addresses in **osicfg.xml** are ignored.

  In the Client sample applications (**cositcps0** and **cositpxs0**), the SCL file doesn't contain a "local" address, so the local address from "osicfg.xml" is used (as before). But any addresses configured in SCL may now be used as the "remote" address.

  This new scheme is enabled at compile time by defining **`USE_DIB_LIST`** in the appropriate makefiles. It may be disabled by removing this definition.

General IEC 61850 MMS Lite Changes

- Added a sample SCL file with CDCs and Logical Nodes from IEC 61850-7-420 (Distributed Energy Resources) Edition 1 (*#2103*).

- Improved Abort request handling (*#2817*).
  If an Abort request is sent (by calling **`mvl_abort_req`**) BEFORE a TCP connect confirm is received, the code now releases the connection resources immediately. In the past, the code would wait for the TCP connect to time out (usually several seconds). This would sometimes make it impossible to establish new connections while waiting for the time out.

- Changed the function **`strncat_maxstrlen`** to return an error if the result is truncated. This allows the calling code to easily check for truncation. Some code should not allow truncation (*#3321*).

New functions (see MMS-Lite Reference Manual for details)

```
scl_parse_cid
scl_parse_scd_all
scl_parse_scd_filtered
scl2_ld_create_all_scd
scl2_vmd_create_all

iecGooseSubscribeExtRefAll
iecGooseUnSubscribeAll
iecGooseSubscriberFind
```

Vulnerabilities

This release of MMS-Lite (V6.0000) corrects the following vulnerabilities:

Note on Klocwork: SISCO uses Klocwork code analysis tools to improve our software products by eliminating common coding mistakes as early in the design process as possible. Each time Klocwork is updated, it can generate warnings related to old code that did not generate warnings in the past. SISCO addresses all the warnings from Klocwork prior to release of any new version. It is possible that these warnings could indicate a problem with the old code that might have been exploitable by an external entity (e.g., a vulnerability). At the same time it would not be appropriate for SISCO to assume that all data passed into each function used in MMS-*Lite* is data that can be injected by an external source and thereby making it a vulnerability. Therefore, we do not classify all Klocwork warnings that we correct as security vulnerabilities. If Klocwork discovers a coding problem that is clearly a vulnerability, we will classify it as such. Lacking specific information that a given warning is externally exploitable we do not classify that Klocwork warning as a vulnerability.

- Fixed several problems detected by the Klocwork source code analyzer, including buffer overflow problems and uninitialized variables (*#3317*, *#3318*, *#3319*, *#3320*).

Known Software Anomalies

Managing Abort and Release indications in Client applications

When an Abort or Release indication occurs, the user function pointed to by **`u_mvl_disc_ind_fun`** is called. If the user does not set this function pointer or the function ignores the indication, it is likely that the connection will be re-used the next time the client establishes a connection. In this case, it would appear to the user application that there are actually 2 connections but both would reference the same MVL_NET_INFO structure. Requests for both connections would be sent to the second server.

To avoid this problem, the user application should be sure to set the **`u_mvl_disc_ind_fun`** function pointer and make sure the function it references manages all connections. Since this function is called from the MVL library and there is no way to pass information back through the library, the best way for the user application to manage connections is to create its own global array of connection pointers, like:

```
MVL_NET_INFO *net_info_array[64];
```

When the user function is called, the user must find the connection pointer in the array, and set the pointer to NULL. When the main application sees that the pointer is NULL, it can assume that the connection was lost.

SMEM

Some issues have been found in the **`smemcfgx`** function used to configure the optional SMEM Pooled Memory Manager. This function parses the **smemcfg.xml** file. If SMEM is used, it is recommended that you simply use the **`m_add_pool`** function directly to configure memory pools. For example, we suggest:

```
 m_add_pool (MSMEM_GEN, 30,  99, SD_TRUE, 50, "pool_99");
m_add_pool (MSMEM_GEN, 20, 499, SD_FALSE, 0, "pool_499");
m_add_pool (MSMEM_GEN, 10, 999, SD_FALSE, 0, "pool_999");
m_add_pool (MSMEM_GEN, 5, 2499, SD_FALSE, 0, "pool_2499");
```

```
  m_add_pool (MSMEM_DEC_OS_INFO, 30,  99, SD_TRUE, 50, "pool_99");
  m_add_pool (MSMEM_DEC_OS_INFO, 20, 499, SD_FALSE, 0, "pool_499");
  m_add_pool (MSMEM_DEC_OS_INFO, 10, 999, SD_FALSE, 0, "pool_999");
  m_add_pool (MSMEM_DEC_OS_INFO, 5, 2499, SD_FALSE, 0, "pool_2499");
```

and so on for each of the 26 SMEM contexts.

Buffer Not Freed on Exit

When terminating the MMS-*Lite* sample client or scl_srvr application, a message is displayed that some buffers are not freed. This only occurs when SLOG IPC is enabled (*#3353*).

Software Corrections

This release of MMS-Lite (V6.0000) contains the following corrections:

IEC 61850 MMS Lite Corrections

- Fixed the SCL file name opened by the sample client application (*#2446*).
  The client now correctly references **"../scl_srvr/sisco_sample.cid"**.

- Corrected the calculation of the BRCB buffer size (*#2852*).
  Previously, the code only counted the size of the data in a report buffer, not including any overhead. When the data was small compared to the overhead, the total memory allocated for a buffered report would be much greater than the configured BRCB buffer size. Now the code accurately counts the total memory used for the report buffer.

- Eliminate unexpected dchg and qchg reports at startup (*#2853*).
  Added a call to a new function **`mvl61850_rpt_data_init`** at startup (after **`datamap_cfg_read`**) to get initial values for all attributes in any Report Dataset. This should avoid triggering reports the first time **`mvl61850_rpt_service`** is called just because the values are changing from 0 (i.e., uninitialized) to a "real" value.

- Fixed the File Directory response for Linux (and any platform that used the POSIX file functions) (*#2930*).
  IEC 61850-8-1 Edition 2 recommends appending a "/" character to directory names. The Conformance Test Procedures appear to be "requiring" this behavior.

- Removed unnecessary **ICCP_LITE_SUPP** define (*#2944*).
  This definition just caused unnecessary confusion.

- Klocwork warning: In **client.c**, in the function **`test_iec_control`**, the variable **`oper_data_buf`** was too small, causing incorrect data to be sent in the Write request. This was fixed by allocating the correct size (*#3283*).

- Fixed the server to return default values for Health in LLN0 if it is present ONLY in LLN0. In IEC 61850 Edition 1, Health is mandatory in all Logical Nodes, but in Edition 2, it is mandatory ONLY in LLN0. Normally, Health in LLN0 is calculated from Health in all other Logical Nodes. If Health is not present in other Logical Nodes, then Health in LLN0 should contain default values (*#3288*).

- Klocwork warning: Fixed mms_adl.c to avoid using a negative value as an array index (*#3303*).

- Fixed the optional **slogl** library to be a compatible replacement for the **slog** library.
  Note: The **slogl** library does not support IPC logging. Do not use it with **slogipc.c** (*#3310*).

Details of Changes in Major Source Files

scl.h

- Added the function **scl2_vmd_create_all**.
- Added **scl_parse_cid**, **scl_parse_scd_all**, **scl_parse_scd_filtered** functions.
- **scl_gse_find**, **scl_address_find**: changed arguments.
- Added **scl_server_find, scl_smv_find, scl_gcb_find,** and **scl_svcb_find** functions.
- Added **ix** to **SCL_FCDA** for Edition 2
- Added **scl_get_attr_ptr, scl_get_attr_copy, scl_get_int_attr,** and **scl_get_uint_attr** prototypes so these functions can be used from other modules.
- Reduced **MAX_FLAT_LEN** to only what is needed (MAX_IDENT_LEN+7). This is just enough for the attribute name plus a 5 digit array index plus brackets.
- **SCL_GCB**: added "subscribed" flag.
- **SCL_INFO**: added the "**SCL_SERVER *serverHead**" element and deleted **ldHead** and **iedName**.
- Added the functions **scl_extref_add**, **scl_server_add**, **scl_ld_destroy**, **scl_dib_entry_save**.
- Added **SCL_OPTIONS** structure to be used in **SCL_INFO**.
- Moved **SCL_DEC_CTRL** from **sclparse.c** to **scl.h**.
- Added the **SCL_SERVER** structure.
- Moved the **brcb_bufsize** attribute from **SCL_INFO** to **SCL_SERV_OPT**.

sclparse.c

- **scl_parse**: now calls the new function **scl_parse_cid** with the **(SCL_OPTIONS *)** argument equal to NULL.
- Added **scl_parse_cid**, **scl_parse_scd_all**, **scl_parse_scd_filtered** functions.
- Changed the SCL storage scheme by using the **serverHead** linked list in **SCL_INFO**. Data for multiple Servers can now be stored. The data for each server is stored in one entry on the **serverHead** linked list.
- Added code to parse the "ix" attribute in the FCDA element for Edition 2.
- Added code to parse the "Inputs" and "ExtRef" elements.
- Fixed Klocwork warnings.
- Removed static from several functions (needed by new modules).
- Moved **SCL_DEC_CTRL** from **sclparse.c** to **scl.h**.
- Added code to save "MinTime" and "MaxTime" from the GSE element.

sclstore.c

- Added the **scl_server_add** function.
- Changed several functions to use the new **scl_info->serverHead** and to store LD information in **scl_info->serverHead->ldHead** instead of **scl_info->ldHead**.
- Added the **scl_extref_add** function to allocate and initialize a **SCL_EXTREF** structure.
- Added the **scl_ld_destroy** function to free all objects from one Logical Device, and call it from a loop in **scl_info_destroy**.

sclproc.c

- **scl2_datatype_create_all**: create types only if needed, loop through ALL servers.
- **scl2_ld_create_all** has been deprecated. Users should use the new function **scl2_ld_create_all_scd**. It is called by the function **scl2_vmd_create_all** (also new). You can see this demonstrated by running the **scl_srvr** sample application and using the command line arguments "**-m scd**".
- Deleted the **reportScanRate** argument from several functions. This parameter is now stored in the **serv_opt.reportScanRateMs** member of the **SCL_SERVER** structure which is passed to those functions.
- Changed several functions to use the new linked list of Server structures **(SCL_SERVER *serverHead**) in **SCL_INFO**.
- Changed several functions to find **iedName** and **ldHead** (linked list of **SCL_LD**) in the **SCL_SERVER** structure, not in the **SCL_INFO** structure.
- Added the functions **scl2_lcb_create_all**, **scl2_app_ref_init** and call them automatically while processing the LogControl elements of SCL.
- Changed to allow **datSet=""** in a LogControl element. In this case, the **DatSet** attribute of the LCB is just initialized to "" (an empty string).
- Moved the definition of **MVL61850_LCB_DATA** to the header file **mvl_uca.h**.
- Added **iedName** and **apName** arguments to the function **scl_address_find**. The function now finds the address where both match.
- **scl_gse_find**: changed all arguments.
- Added the function **scl_gcb_find**.
- Moved the **scl_svcb_find** function from **scl_srvr.c** to **sclproc.c** and changed the arguments.
- Added the function **scl_smv_find**.
- Added the function **scl_server_find**.
- Added the function **scl2_datatype_count_scd** to count the number of MMS types needed.
- Added the (**SCL_SERVER ***) argument to several functions.
- Added the argument **is_client** to more functions. When **is_client==SD_TRUE**, the functions **DO NOT** initialize data based on DAI or control block elements of SCL.
- Fixed all Klocwork warnings, mostly by using **strncat_maxstrlen**.
- Added FC = SR, OR, and BL for IEC 61850 Edition 2.
- Added bType = TrgOps, OptFlds, SvOptFlds, PhyComAddr for IEC 61850 Edition 2.
- Added bType = EntryID for IEC 61850 Edition 2.
  Note: EntryID is **NOT** defined in 61850-6 but it is needed for IEC 61850-7-2.
- Added **ResvTms** & **Owner** attributes for BRCBs; added **Owner** for URCBs (for IEC 61850 Edition 2 only).
- Added the following functions and structures for IEC 61850 Edition 2:

```
ms_rt_bld_61850_add_brcb_ed2ms_rt_bld_61850_add_urcb_ed2
MVL61850_GCB_DATA_ED2
ms_rt_bld_61850_add_gcb_ed2
MVL61850_MSVCB_DATA_ED2
ms_rt_bld_61850_add_msvcb_ed2
MVL61850_USVCB_DATA_ED2
ms_rt_bld_61850_add_usvcb_ed2
```

- Added the **PHYCOMADDR** structure for storing GOOSE and SampledValue addresses, and changed functions to use it. Added the function **ms_rt_bld_61850_add_phycomaddr** to use when constructing type definitions containing these addresses.
- Changed the **SmpRate** attribute in MSVCB & USVCB from INT32U to INT16U for both Edition 1 and Edition 2 (IEC 61850-9-2 just says type is "Integer" but IEC 61850-7-2 says INT16U).
- Added code to read the "count" attribute from SDO elements in SCL and pass it to **rtadd_sdo**. Used the "count" to create arrays of objects.
- Changed **scl2_ld_create_part1** to make sure the domain created ALWAYS has enough room for the objects defined in SCL.
- In **scl2_dai_set_value**, changed the **prim_num** from **ST_RTINT** to **ST_UINT** (need bigger values for Edition 2 MHAI).
- Changed **scl2_dataset_create** to allow "ix" attr in FCDA (for Edition 2 only).
- Removed statements to disable **MVLLOG_NERR** before calling **scl2_dataset_create** (it is helpful if there are problems).

client.c

- Added the **USE_DIB_LIST** option (enabled by makefiles).

scl_srvr.c

- Moved the **scl_svcb_find** function to **sclproc.c** (in the "mvlu" library) and changed its arguments.
- Added a call to **startup_cfg_read** near the top of **main**. This information is needed before anything else.
- Changed the third argument (**max_rt_num**) passed to **scl2_datatype_create_all** to 0 (i.e. no limit on the number of **RUNTIME_TYPE** structures used in each type definition).
- Added functions **u_mvl_jinit_ind**, **u_mvl_jread_ind**, and **u_mvl_jstat_ind**.
- Added a call to **mvl61850_log_service** in the main loop. This is required to monitor Log events and to automatically generate Log entries.
- Added a call to **mvl61850_log_ctrl_destroy_all** on exit to free up resources allocated for any configured IEC 61850 Logs.
- If the SCL file indicates Edition 2, then the SampledValue messages are encoded according to Edition 2.
- Added the **USE_DIB_LIST** option (enabled by makefiles).
- Added the '-m' command line argument to demonstrate new functions.
  If "-m scd" is entered on the command line, the following new functions are called:
  scl_parse_scd_all
  scl2_vmd_create_all
  iecGooseSubscribeExtRefAll (MMS-LITE-802-001 version only)
- Added the function **all_obj_destroy_scd**.
- Added the function **init_etype_info**.
- Added the function **mvl61850_rpt_data_init** to set appropriate initial values for all Report Datasets.
- Added a call to **subnet_serve** in main loop (to receive GOOSE messages).
- Added a call to **datamap_cfg_log** to log all data mapping information. This call is disabled with #if and #endif directives, but it may be enabled for debugging purposes.
- Added a call to **scl_log_all** after all SCL parsing. This will log all SCL information stored by the parser, but only if **SX_LOG_DEC** is enabled in **logcfg.xml**.
- Added #if and #endif directives around each indication function so they are compiled only if the appropriate service is enabled in **mmsop_en.h**.
- Fixed the name of **mvl61850_ctl_direct_enhanced_get** (previously the name contained "61860").

General Application Upgrade Notes for MMS-EASE *Lite*

MMS Lite TP4 Support Removed

Support for the Stack using OSI Transport Class 4 (TP4) has been removed from the product. Most customers are not using TP4, so this change should have little impact on current applications. However, because of the way the different layers of the stack worked together, if your application performs MMS communication and also receives IEC 61850 GOOSE, GSSE, GSE Management, or SampledValue messages, some simple application changes are required. These changes are necessary because previously the application only needed to call **`mvl_comm_serve`** to receive all incoming messages. This would call through the TP4 layer to the **`clnp_snet_read`** function in the "Subnetwork" interface to receive raw Ethernet packets. This function could only be called from one place, so the processing of GOOSE, GSSE, etc., messages had to be coordinated with the TP4 layer. With the TP4 layer removed, this is no longer possible. Fortunately, the new approach is easy to implement and is actually much more efficient (i.e., unnecessary TP4 processing is eliminated). There were many changes in the libraries to eliminate TP4, but only the following changes should be important for user applications:

- TP4 support was removed by eliminating the **ositp4e** library and changing the **ositpxs** library by removing the define **`TP4_ENABLED`** from the makefiles. The following source files were removed:
  clnp_thr.c
  tp4_dec.c
  tp4_hc.c
  tp4calld.c
  tp4callg.c
  tp4main.c
  tp4send.c

  The makefiles for the sample applications "iecgoose" and "gse_mgmt_test" were changed to link to the "ositpxs" library. If you have ported the makefiles, you will need to make similar changes.

- Previously the functions **`clnp_init`** and **`clnp_end`** were called automatically from the TP4 layer. With TP4 removed, they must be called directly from the user application. This is only necessary if the application sends or receives GOOSE, GSSE, GSE Management, and Sampled Value messages.
- A new function **`subnet_serve`** was added to receive and process GOOSE, GSSE, GSE Management, and Sampled Value messages without using TP4. You may easily modify this function to receive only the packets you want by defining any combination of the following: **`GOOSE_RX_SUPP`**, **`GSSE_RX_SUPP`**, **`GSE_MGMT_SUPP`**, **`SMPVAL_SUPP`**. You can also enable code useful for threaded applications by defining **`SUBNET_THREADS`**.
- The client sample application was modified to demonstrate the use of **`subnet_serve`**. It now calls **`clnp_init`** and **`clnp_end`** as explained above. The call to **`clnp_snet_read_hook_add`** and related code was deleted. See **client.c**.
- The GSE Managment sample application was modified to use **`subnet_serve`** function. It now calls **`clnp_init`** and **`clnp_end`** as explained above. The call to **`clnp_snet_read_hook_add`** and related code was deleted. See **gse_mgmt.c**.
- The IEC GOOSE sample application was modified to call **`clnp_init`** and **`clnp_end`** instead of **`mvl_start_acse`** and **`mvl_end_acse`**. This eliminates the need for unnecessary MVL, ACSE, and stack code. This could also be modified to use **`subnet_serve`** like the other samples, but that does not appear necessary right now. See **iec_main.c**.
- The uca_srvr sample application was modified to eliminate the code to receive GSSE messages. The code for receiving GSSE messages may now be added to any application by just calling **`subnet_serve`** and enabling the GSSE code.
- Pointers were added to the **`SN_UNITDATA`** structure so that these structures can easily be put on a linked list. This may be useful in threaded applications where one thread receives packets by calling **`clnp_snet_read`**, determines the type (GOOSE, GSSE, etc.), and stores them on linked lists. Then other threads can get them from the linked lists to process them later. This approach is demonstrated in **subnet.c**.

- The function **`clnp_snet_read_hook_add`** was eliminated. This was used to allow receiving of GOOSE, GSE Management, and Sampled Value messages through the TP4 layer. With TP4 eliminated, it is no longer useful. Instead, the new function **`subnet_serve`** must be used to receive these messages.
- The function **`clnp_read_thread_start`** was eliminated. This was used in some older applications for receving GSSE messages and TP4 messages. One thread received packets and passed them up to two separate threads for TP4 and GSSE. With TP4 eliminated, it is no longer useful. Instead, the new function **`subnet_serve`** must be used to receive GSSE messages.
- The functions **`a_get_event_handles`** and **`a_get_event_handles_unix`** were eliminated. These functions returned different handles depending on whether TP4 was enabled. With TP4 eliminated, they are no longer useful. The **`wait_any_event`** function in **event.c** was simplified to avoid using these functions. Users who modified **`wait_any_event`** for their application may need to modify it again.
- The undocumented functions **`clpp_initialize`**, **`clpp_terminate`**, and **`audt_set_msgsize`** were eliminated. The GSSE code was simplified so these are no longer needed.

Changes and Corrections in V5.3001, V5.10 and V5.06

MMS-EASE *Lite* (V5.3001) contained the following changes:

- Added the following files to the stack library:
  acse2dec2.c
  acse2dib_arr.c
  copp_dec2.c
  coppdcpa.c

  Deleted this obsolete file from the library:
  lean_cfg.c

- Simplified the process to enable MLOG logging (i.e., the logging of MMS primitives in an easy to read form). You only need to define **`MLOG_ENABLE`** in the makefile when compiling **mmsop_en.c** (see **cositcps0.mak**, for example). The function **`ml_mlog_install`** is no longer needed to initialize this feature, so it does not need to be added to any sample or user application. It was replaced with an empty macro (for backward compatibility only). This function just initialized several global function pointers. Instead, all these function pointers were moved into one global structure **`mlogl_info`**. This is initialized automatically at compile time (*#2364*).

- Changed all Bind IDs and Connection IDs in the stack library from **`ST_LONG`** to (**`ST_VOID *`**). Look in the source code for **`bind_id`** and **`conn_id`**. These variables contain pointers which were previously cast to ST_LONG, but the casting was not legal on some newer systems (especially 64-bit systems).

- Added these new functions to provide a more simplified ASN.1 decoding scheme:
  `asn1r_decode_init`
  `asn1r_get_tag`
  `asn1r_chk_tag`
  `asn1r_get_contents`
  No function pointers are needed. These functions were used to simplify the decoding of ACSE and COPP (Presentation Layer) PDUs.

- Deleted these global variables:
  `_mmsdec_rslt`
  `_mms_dec_info`
  `_mms_dec_info_pres`
  The new **`mmsdec_rslt`** member of the **`ASN1_DEC_CTXT`** structure is used instead.

- Increased flexibility in selecting the Ethernet interface in Linux by using the **`<Network_Device>`** field in **osicfg.xml**. Previously, this was only used on Windows. If the **`<Network_Device>`** field is not present or empty, the default device “eth0” is used (*#1220*).

- Eliminated “binding” for “calling” connections. Instead of binding to a Local address at startup, the user must simply pass a Local AR Name (new argument) to **mvla_initiate_req_ex**. The Local AR Name is used to find the Local address in the configuration. This change allows any “calling” application to use multiple Local addresses by simply passing different Local AR Names to **mvla_initiate_req_ex**. (*#2290*). Please refer to *General Application Notes* section of this document.

  NOTE: Binding is still required for “called” connections.

- These alternative connection functions were deleted:
  mvl_initiate_req
  mvla_initiate_req
  All applications must now call **mvla_initiate_req_ex** to establish “calling” connections (*#2290*). Please refer to General Application Notes for MMS *Lite* section of this document.

- Added support for the SISCO Message SEAL. The RPL Security Seal system provides an authentication and data validation mechanism, using private key encryption, CRC data validation, and anti-playback timestamps. The RPL Security Seal mechanism provides improved protection for the RPL communication channel and is intended to be used to make it much more difficult for nefarious external parties to compromise or exploit this communication channel. (*#1265*).

- Deleted all “synchronous” client functions (e.g., **mvl_identify**, **mvl_status**, **mvl_getnam**, etc.). They provided little benefit and increased code size. They can easily be replaced with a call to the corresponding “asynchronous” function followed by code to wait for the response (see **waitReqDone** in the sample **client.c**). The function pointer **u_mvl_check_timeout** (used by these functions for timing out) was also deleted.

- Changed the MMS **Write** response processing to save the **WRITE_RESULT** information from the Write Confirm in the **MVL_WRITE_REQ_INFO** structure so the user application can access it. Changed the MMS **Read** response processing to save the **ACCESS_RESULT** information from the Read Confirm in the **MVL_READ_RESP_PARSE_INFO** structure so the user application can access it (*#1596*).

- Modified the **DIB_ENTRY** structure slightly to be compatible with other SISCO products. Moved some **DIB_ENTRY** related functions to new modules **acse2dib_arr.c** (used if the **DIB_ENTRY** is stored in an array) and **acse2dib_ll.c** (used if the **DIB_ENTRY** is stored in a linked list).

  IMPORTANT: Users of the **HARD_CODED_CFG** option must make small changes when initializing the **DIB_ENTRY** tables. The **HARD_CODED_CFG** code was deleted from all sample applications except the client so use **client.c** as a reference.

- Changed the GNU makefiles (for Linux, QNX, etc) to define INCDIR and LIBDIR *only* in **platform.mak** (easier to change the definitions if needed).

- Changed the “foundry” utility to generate the source code to initialize the **type_ctrl** member in each **MVL_VAR_ASSOC** structure (i.e., in the variable definition). Applications can now assume this attribute is valid for ANY variable, (i.e., for “dynamic” or “foundry generated” variables).

- Deleted all Security code from **tp0_socks.c**.

- Deleted these files:
  ssecusr0.c
  ssec0.mak
  ssec0.vcproj

  The file **ssecusr0.c** contained only "stub" functions to simplify adding security later but it is no longer needed. The other 2 files were used to compile it and create the **ssec0** library.

- Changed the GNU shell scripts to create object file directories only if they don't already exist.

- Added the "-m32" compiler option on LINUX to make sure everything is compiled for a 32-bit environment.

- Added these files to the utility library:
  crc_util.c
  ipc_utl.c
  sseal.c

- Deleted support for the obsolete "Reduced Stack" (enabled by defining **REDUCED_STACK**), the "Trim-7 Stack" (enabled by *not* defining **MOSI**), and code enabled by defining **MAP30_ACSE** (was used in other SISCO products). By default now, the code that was enabled by defining **MOSI** is always used (you don't need to define **MOSI** to enable it).

- Replaced **cosp_enc_cn_ac** with 2 separate functions: **cosp_enc_cn** to encode a Connect PDU and **cosp_enc_ac** to encode an Accept PDU (simpler and clearer).

- Deleted the obsolete function **sx_parseEx_mt**. All applications must now use **sx_parseExx_mt**.

- Moved the **u_custom_rd_ind** function to **userleaf2.c** (new file). This should make it easier to customize or replace.

- In the **findalgn** utility, used to generate alignment information for the target compiler, corrected the value generated for **STREND_ALGN** if the "structure end" mode is **M_STREND_MODE_NONE** (see **findalgn.c**).

- Removed all global variables used by **ms_asn1_data_to_runtime** and several lower level functions (see **mmsdatat.c**). Instead, the same data is stored in a local **USR_INFO** structure which is passed to all lower functions. This change makes the code thread safe, so the mutex semaphore was also removed.

- Increased the maximum size of "selectors" to be compatible with other SISCO products. The following defines are in **acse2usr.h** and **tp4api.h**:

  ```
  #define MAX_PSEL_LEN 16
  #define MAX_SSEL_LEN 16
  #define MAX_TSEL_LEN 32
  ```

  If necessary, you may change these to smaller values to reduce memory usage. Just be sure to recompile everything (*#2230*).

- Changed the recommended version of WinPcap to V4.1.2 (the latest version available for download from www.winpcap.org). See the instructions for using WinPcap in the Reference Manual. Other versions might work just as well for you, but this is the version tested for this release (*#2334*).

- Simplified the procedure to use the Expat XML parser. Follow the procedure in *Using Expat* in the *General Application Notes* section of these Release Notes **instead** of the instructions in the Reference Manual.

- Support for the Stack using OSI Transport Class 4 (TP4) has been removed from the product. Most customers are not using TP4, so this change should have little impact on current applications. Please refer to *MMS-Lite TP4 Support Removed* in the *General Application Notes* section.

- All files are now installed with correct lower case names so they can easily be transferred to Linux or UNIX systems without converting to lower case names. This should allow customers to use almost any FTP client to transfer the files (*#2415*).

  NOTE: Be sure to still use ASCII transfer so that the CR/LFs are correctly converted.

IEC 61850 MMS Lite Changes

- Overhauled **datamap_cfg_read** (in **usermap.c**) to greatly speed up the processing of large **datamap.cfg** files by using the **qsort** and **bsearch** functions. The **ldevice** (Logical Device) member of the **DATA_MAP** structure was removed. Since there are many **DATA_MAP** entries for the same Logical Device, there were many duplicate "ldevice" strings. Instead, "ldevice" is stored in the **DATA_MAP_HEAD** structure allocated for each Logical Node. This significantly reduces memory usage. If "ldevice" is needed in any leaf function, it can easily be accessed (see **map_head_sorted** in **u_custom_rd_ind**) (*#1324*).

- IEC 61850 deadband calculations are now automatically performed in the server. As the **datamap.cfg** file is read, deadband information is saved in **`map_entry->deadband_info`**. This information is used later to automatically perform deadband calculations for attributes like **mag**. If you DO NOT want the automatic deadband calculation for a particular attribute, you can override it by simply mapping the attribute in **datamap.cfg**. For example, if you map **MMXU1$MX$TotW$mag$i** in **datamap.cfg**, your customized leaf function must perform the deadband calculation.
  Added these files to the **scl_srvr** sample application to perform IEC 61850 deadband calculations automatically:
  db_61850.h
  db_61850.c
  userleaf2.c

- Added the **`SCL_ADDRESS`** structure to store addressing information from the "Address" element of the SCL file (inside the "ConnectedAP" element). This structure is accessed using pointers in the main **`SCL_INFO`** structure and it is filled in by the **`scl_parse`** function.

- When parsing the SCL file, allow the optional attribute **gi** in the **TrgOps** element, and set its default value to **true** (required by IEC 61850-6). This feature is important for some users because if the Client was required to write the **TrgOps** attribute of a Buffered Report Control Block to set the **gi** bit, this would cause an automatic Purge of the buffer (and loss of data) (*#1952*).

- Fixed the handling of EntryID according to IEC 61850 Tissue #278 "EntryID not valid for a server" (*#876*).

- Allow the basic types INT24 and INT23U to be configured in SCL, and automatically convert them to INT32 and INT32U in the MMS model (*#2068*).

Vulnerabilities

MMS *Lite* (5.3001) corrected the following vulnerabilities:

- Fixed memory leaks that could occur when **`mvla_fget`** is called or when an ObtainFile indication is received (see **`mvlas_obtfile_resp`**), if the subsequent "automatic" calls to **`mvla_fopen`**, **`mvla_fread`**, or **`mvla_fclose`** failed (*#2062*).

- Fixed memory leaks that occurred under the following error conditions:
  when a MMS Reject PDU is sent after the decoding of a MMS Confirmed Response fails (*#2053*).
  when a MMS Reject PDU is received for a Confirmed Request but the Invoke ID does not match any outstanding request (*#2054*).
  when a MMS Conclude Error PDU is sent (*#2063*).
  when a MMS Conclude Error PDU is received (*#2065*).

- Corrected a memory leak that occurred in **gensock2.c** (*#2075*).

- Corrected a vulnerability that caused a DoS when receiving an AARQ message with the user-information data sent as octet-aligned rather than single-ASN.1-type.

- Made corrections in several files based on static code analysis. These include checking for NULL pointers and ensuring buffer overflow can never occur (*#1902*).

- Fixed a crash that could occur decoding an ACSE Authentication password if the password was too long (*#2175*).

- Corrected a problem when "osicfgx" is called a second time (to reconfigure addressing) that caused writing of data to a buffer that was already freed. This could corrupt other data or corrupt the heap (*#2207*).

- Corrected a problem that caused a crash in **s_fdir.c** when MMS logging is enabled and an application is responding to a File Directory indication. (*#2243*).

- Fixed a memory corruption problem that could occur when a Read or Write indication is received with Alternate Access on an array. If the array index was out of range, memory after the array would be accessed (*#2490*).

Software Corrections

MMS-Lite (V5.3001) contained the following corrections:

- The software now correctly parses unsigned integers from XML files. The parser functions **sx_get_uint, sx_get_uint16**, and **sx_get_uint32** do not allow negative numbers or numbers with a decimal point (*#788*).
- Corrected the computation of the number of requests outstanding. In certain cases, it was not being decremented when it should have been, so the application could not send more requests (*#2055*).
- Corrected the function, **mvlu_resv_wr_ind_fun** in **mvlu_rpt.c** so that the **Resv** is not writable if RptEna = TRUE. (*#2233*).
- Used a more portable way to clone an **OBJECT_NAME** structure (see **mvlu_clone_objname** in **mvl_uca.c**). This should align the data correctly for any CPU and compiler.
- Fixed the alignment calculations for variable-length Bitstring and variable-length Octetstring when generating the **RUNTIME_TYPE** definition (see **mms_tdef.c**). These are defined as “structures” in the "C" code representation of the data, so they must be aligned in memory as "structures". This change will have no impact on some systems like Windows where these structures were aligned the same as **ST_INT16** (16 bit integer), but will help on systems that have special rules for aligning structure (*#1801*).
- Corrected the return value for **a_get_event_handles**. It must be (ST_EVENT_SEM *).
- Fixed the format string or argument types for many log messages (*#1835*).
- Fixed the type creation code to allow UTF8Strings longer than 32767. An incorrect cast in mms_tdef.c caused truncation if (el_len > 32767). Before this fix, an error message was logged (Unicode UTF8 string too long) and the type creation failed (*#2336*).
- Corrected the SCL parser to allow up to 65 characters in the **rptID** attribute of a **ReportControl** element (as required by IEC 61850-7-2). Previously only 64 characters were allowed (#2371).

IEC 61850 MMS Lite Corrections

- Corrected the computation of the length of “flattened” IEC 61850 variable names, to support the maximum allowed length without buffer overflow (see **mvlu_rt.c**).
- When parsing SCL files, fixed the conversion of APPID, VLAN-PRIORITY, and VLAN-ID elements in GSE and SMV addresses to correctly interpret the data as base 16 (i.e., hexadecimal) values (*#1224*).
- Added checks for invalid characters in SCL files. Previously, invalid characters in several elements could cause invalid MMS objects to be created, or cause object creation to fail (*#2249*).
- Fixed the SCL parser to handle Unicode string initial values (in **Val** elements inside **DAI**, **DA**, or **BDA** elements) and correctly initialize the MMS data. This was corrected in **sclproc.c** (*#1383*).
- Fixed the “reason” sent in some IEC 61850 reports when multiple reasons are detected by the server. Only reasons that are enabled in **TrgOps** will be sent in reports (*#2381*).
- Fixed the error responses for some IEC 61850 requests:
  If an unknown data set is written to **DatSet** in an RCB, a response is sent with DataAccessError=object-value-invalid (11).
  If **EntryID** is written to a BRCB and it is not found in the buffer, a response is sent with DataAccessError=object-value-invalid (11).
  If a CreateDataSet request is received and the DataSet already exists, an error response is sent with error class=DEFINITION (2), error code=OBJECT-EXISTS (5).
  If a CreateDataSet request is received and the Domain is not found, an error response is sent with error class=DEFINITION (2), error code=INVALID-ADDRESS (2) (*#2382*).
- Fixed problems detected by a static code analyzer in SCL processing code (**sclproc.c**) (*#2383*).

- Eliminated “assert” failures in **mvl_uca.c** when a MMS Read indication is received with an Alternate Access Selection type of “indexRange” or “allElements”. Instead, an Access Result error will be returned in the MMS Read response (i.e., this type of Alternate Access is not supported) (*#2398*).
- Changed the Access Result Error returned when IEC 61850 Control operations fail to OBJECT-ACCESS-DENIED (3), as specified in IEC 61850-8-1 Edition 1 Table 72 (*#2421*).

MMS-EASE *Lite* (V5.10) contained the following changes:

1. *MMS-Lite Reference Manual* has been significantly modified.
2. SISCO has dropped support for the OSILLC driver and adopted the **WinPcap** solution for **Windows** applications. WinPcap is basically a Windows portation of the well known libpcap API on Unix. WinPcap is not present on the MMS Lite CD. There are instructions for downloading WinPcap for the 802 version of MMS-EASE Lite in the *MMS-EASE Lite Reference Manual*. Without WinPcap, the 802 sample solutions for TPX, GOOSE, GSSE and Sampled Value will not build. Unless these steps are taken, only the TCP version of MMS Lite will compile. If you are using MMS-LITE-801-001, it only has support for TCP so WinPcap is not needed.
3. To allow configuration of the network device needed by the **WinPcap** API, the **`network_device`** element was added to the **`CLNP_PARAM`** structure. This parameter is initialized by the **<Network_Device>** element added to the **osicfg.xml** file.
4. A new Windows utility called the **PCAPEthernetConfiguration** was added to the MMS-LITE-802-001 product. It configures the Ethernet Adapter needed by the WinPcap API described above. It allows the user to select a local Ethernet Adapter from a list and puts that selection in the **osicfg.xml** file. Please review the XML encoded tag <Network_Device> in **osicfg.xml** to be sure its value accurately reflects an Ethernet adapter present on the actual PC. There are a number of **osicfg.xml** files in MMS Lite sample directories. The OSI, GOOSE, GSSE, and Sampled Value capable samples all need to have the **PCAPEthernetConfiguration Utility** run on their **osicfg.xml** files. The utility is found in the **\mmslite\bin** folder. The PCAPEthernetConfiguration Utility can be executed from:

   Start→All Programs→SISCO→MMS-EASE Lite→PCAP Ethernet Configuration Utility

5. A change was made to header file **sysincs.h so** that **winsock2.h** is included before **windows.h**. It was found that **winsock2.h** must always be before **windows.h** to override defines in **winsock.h** (included by **windows.h**). Since all MMS-EASE Lite source modules include **sysincs.h**, this is the best place to include winsock2.h.
6. A change was made to the code that reads in the files **startup.cfg** and **datamap.cfg** in the **\scl_srvr** folder. The change is to only accept tabs and commas as delimiters (not spaces) and to treat extra tabs or commas as blank fields. Previous copies of the **startup.cfg** file will no longer work because they contained extra tabs. The current **startup.cfg** is not overwritten by the MMS Lite installation. This means if you are upgrading from an older version of MMS Lite, the **startup.cfg** file will have to be modified with a text editor to remove extra tabs or there will be errors reading the file. This change was done so that **startup.cfg** and **datamap.cfg** can easily be edited using Microsoft Excel.
7. The SCL parser now converts the text from the **`<EnumVal>`** elements into integer values, so you can use the text when specifying initial values.
8. Sample Client code has been changed to abort the connection after a user timeout. Normally, if a Client tries to connect to a device that is not online, eventually the stack will time out, return an error, and will clean up. If an application does not wait for the stack to time out by setting a shorter “user” timeout, it is possible for the application to attempt another connection before the stack is ready, often causing the following error:

   ```
   2008-03-14 10:58:02.921 MVLLOG_ERR (mvl_acse.c 935)
    Error - Could not get calling MVL_NET_INFO
   ```

   The problem is that the Client needs to abort the connection after the “user” timeout to free up resources immediately. The abort is not automatically done by the stack because the stack did not detect the timeout. The new Client sample code aborts the connection in this case. Most likely, other Client applications need a similar change.

9. The **mvl61850_ctl_chk_state** function was changed to directly set the **AddCause** and the **Error** members of the **LastApplError** for each error condition (*#714*).
10. This version will not allow a client to delete a NVL if any Report Control Block (RCB) contains a DatSet that references the NVL. The client must first change the DatSet in the RCB before deleting the NVL.
11. If the sboClass="operate-many" (i.e., 1) in the SCL file, the Server will now allow many Operate requests without waiting for another Select (reading of SBO or writing of SBOw). Once the control is selected, it will stay selected until a Cancel is received (i.e. the "Cancel" struct is written) or until **sboTimeout** times out.
12. Reduced the maximum memory usage during SCL processing even further (better than V5.0602) by using a 2-pass method for type creation. The function **ms_rt_bld_start** and other **ms_rt_bld*** functions are called twice, first to compute the size of the type, then to actually create the type. Between the two calls, the required memory is allocated. There is no need to allocate extra memory and no need to reallocate later.
13. The following new files have been added:

    sisco_sample.cid (Replaces scl.xml. Also modified **startup.cfg** to reference this file)
    sisco_sample_wind.cid
14. The IEC 61850 Buffered Reporting code has been modified to start buffering on startup as long as the BRCB is configured correctly with a real dataset (i.e., DatSet references a valid dataset). This leads to an increase in memory usage upon startup as the buffer is being filled. This behavior is required by Edition 2 of IEC 61850-7-2. If this behavior is ***NOT*** desired, a bit of code in **mvl61850_create_rpt_ctrl** must be disabled or removed.
15. The function that processes the **datamap.cfg** file (**datamap_cfg_read**) was modified to ignore illegal mappings and log an error. Certain leafs are automatically mapped by the libraries, so there is no need for mapping here (e.g., all RCB leafs and ..$Beh$stVal).
16. The SCL parser makes sure the length of lnClass, prefix, and inst are within allowed limits.
17. Added functions to allow encoding IEC 61850 GOOSE messages in 2 parts (see gsei_enc.c). The “data” part of the GOOSE message does not change with each retransmission, so it can be encoded just once, but the “header” part of the GOOSE message must be re-encoded with different values for each retransmission. These functions may be used to slightly improve efficiency in some applications.
18. Added gensock2.c to the “util” or “utility” library and deleted it from all the “osi*” libraries. This allows any application to use slogipc.c (which requires gensock2.c).
19. Allow the following new **bType** values in the SCL file: **ObjRef** (convert to 129-character visible string), **INT64**, and **INT64U**.
20. Several improvements were made to the MVL object destroy functions to make sure that objects cannot be destroyed when they are still referenced (see details in **mvl_obj.c**). The destroy functions will now fail if an object is still in use and appropriate errors will be logged.
21. All optional code in **mvl_acse.c** that was enabled by defining **OBSOLETE_AA_OBJ_INIT**, has now been removed. This code is no longer supportable. This define must not be used.
22. Removed the makefiles and vcproj files for building the **ositcpe** and **ositpxe** libraries and any executables that link them. Projects must use the newer **ositcps** or **ositpxs** libraries.
23. On Windows, changed the name of the utility library from “**util**” to “**utility**” to reduce conflicts with libraries used in other SISCO products.
24. Added code to parse the **SettingControl** element in the SCL file and to automatically generate the SGCB structure under the SP functional constraint.
25. Deleted the **CALLED_ONLY** and **CALLING_ONLY** defines. These defines were used to reduce the size of code compiled for applications that had only Called connections or only Calling connections, but the savings were minimal. Now all applications include Calling and Called code.
26. Simplified the code in **u_set_all_leaf_functions** so it should be easier for users to customize. Also, fixed the mapping of leafs inside the **setMag** structure.

27. Added the function **mplas_concl_err** to send an error response to a Conclude indication. Modified some sample applications to call it if requests are still pending or if files are still open.
28. Automatically close all open files on an Abort indication.
29. Improved the IEC 61850 Integrity report generation. The new function **mvl61850_integrity_timeout** is called to build the report without scanning for new data because the data was already scanned at the start of **mvl61850_rpt_service**.
30. The logging configuration masks used in the "iecgoose" sample application are now configured using the **logcfg.xml** file processed by the **logcfgx_ex** function.
31. For IEC 61850 Edition 2, changed the Read response for the **EntryID** and **TimeOfEntry** attributes in a BRCB, to return different values depending on the value of **RptEna**.
32. For IEC 61850 Edition 2, if an octet string of all zeros is written to the EntryID attribute of a BRCB, the server starts sending the OLDEST entries in the report buffer.

MMS-EASE *Lite* (V5.0602) contained the following changes:

1. Reduced the maximum memory usage significantly by reallocating some temporary buffers to a smaller size during SCL processing.
   NOTE: V5.10 does even better by avoiding the reallocation altogether.
2. Changed some members of the **RUNTIME_TYPE** structure to **ST_INT** so that Octet Strings and Visible Strings longer than 32K can be configured.

MMS-EASE *Lite* (V5.06) contained the following changes:

1. For Windows – MMS-Lite now uses Microsoft Visual Studio .NET version 2005. All MMS-Lite Samples have been converted from Visual C++ V6.0 workspaces (dsw) to Visual Studio .NET 2005 solutions (sln).
   Note that libraries from this product release cannot be linked with applications using Visual C++ V6.0 or Visual Studio .NET 2003 compiler.
2. Added support for IEC 61850 Sampled Values (see later on in the document for details). Test code may be enabled in **client.c** and **scl_srvr.c** by defining **SMPVAL_SUPPORT** in the makefiles.
3. Added support for IEC 61850 GSE Management (see the section *IEC 61850 GSE Management Data and Functions* for details).
4. Deleted the unimplemented **gs_wait_mult_event_sem** function on UNIX, so that problems are detected at compile time, not at runtime.
5. Improved Foundry so that the initialization code generated for NamedVariableLists completely resolves all Variables. This improves efficiency because there is no need to resolve the Variables each time the NamedVariableList is accessed.
6. Added more **mvlu_get_leaf_*** utility functions (see **mvl_uca.c**). These are used by the MVL library and the **scl_srvr** sample application.
7. Added code to parse and process the "SampledValueControl" and "Communication" elements of an SCL file.
8. The **gse_iec_control_create** function was modified to allocate and copy the caller's strings, so the caller does not need to make sure the strings persist for the life of the GOOSE message.
9. The functions **clnp_snet_write** and **clnp_etype_write** were replaced with a simpler, more flexible function, **clnp_snet_write_raw**. The function was ported to Windows and LINUX, but users will need to port it to other platforms. Users should look at **clnp_linux.c** to see the changes for LINUX.
10. Added the option to use the "Expat" XML parser by simply defining **USE_EXPAT** when compiling **sx_dec.c** and linking to an appropriate "Expat" library. The "Expat" parser seems to be much better at detecting errors in XML files and dealing with unexpected whitespace characters. Please refer to the *MMS-Lite Reference Manual* for detailed information.
11. The "Expat" library is not included with the product, but it should be easy to download version 2.0.1 of the "expat" package from http://expat.sourceforge.net/. On Windows, it should be easy to integrate it with the "Visual Studio 2005" solution – please refer to the *MMS-Lite Reference Manual*.

12. Changed the SCL processing to create multiple RCBs for each ReportControl (Buffered or Unbuffered) in the SCL file. The number of RCBs depends on the value of the "max" attribute of the "RptEnabled" element. This change makes it possible to allow each client to change the DatSet element as needed.
13. Changed **`ObjectReference`** attributes of some of the **`DatSet`** and **`LogRef`** to Vstring129.
14. Added new ASN.1 encoding functions: **`ms_local_to_asn1_2`**, **`ms_adl_to_asn1_2`**, and **`ms_aa_to_asn1_2`**. These are very similar to the existing functions **`ms_local_to_asn1`**, **`ms_adl_to_asn1`**, and **`ms_aa_to_asn1`**, but they are easier to use in most cases.
15. IEC 61850 Reporting changes:
    For IEC 61850 Reporting, the new functions **`mvl61850_create_rpt_ctrl`**, **`mvl61850_free_rpt_ctrl`**, and **`mvl61850_rpt_service`** ***must be used***. The older functions **`mvlu_create_rpt_ctrl`**, **`mvlu_free_rpt_ctrl`**, and **`mvlu_rpt_service`** now work ***only for UCA Reporting***.
    a. Buffered reports are now segmented only when sending, so that changing PDU size does not cause a problem.
    b. The **SqNum** attribute is reset to 0 when a BRCB is enabled. It is only incremented when a report is sent, so that if the BRCB is disabled, **SqNum** will NOT increment, even though reports are still being buffered.
    c. The Report buffer is purged when **DatSet**, **TrgOps**, **IntgPd**, **BufTm**, or **RptID** is written, but only if the value is changed (for Tissue #322).
    d. A BRCB is now reserved only when **RptEna** is written.
    e. Attribute types needed for reporting are now automatically created if not found.
    f. If the Data or Quality Changes are being saved waiting for the **BufTim** to expire, they will be published before the Integrity Report is sent. All saved changes are reported prior to the Integrity report. This prevents Data and Quality changes from being received out of order. This applies to Buffered as well as unbuffered Reports.
    g. If the **RptID** in the RCB is NULL, a **RptID** is automatically generated to send in Reports.
    h. Writing the GI attribute is allowed only if the RCB is enabled. A GI Report is generated only if the general-interrogation bit in **TrgOps** is 1.
    i. Writing the **DatSet** attribute is now allowed and it causes the new data set to be used when generating subsequent reports.
    j. Changed the Report Buffering scheme to save "raw" data in the buffer and encode reports only when sending, so that **SqNum**, etc., can be set/changed each time a report is sent.
    k. The BRCB **EntryID** is automatically initialized with the current time in the first 4 bytes and 0 in the last 4 bytes, so that after a reboot, the initial EntryID is almost always unique. If you want a different initial **EntryID**, you can call **`mvl61850_brcb_entryid_init`** after **`mvl61850_rpt_ctrl_create`** (see usage example in **uca_srvr.c**). The value in the last 4 bytes of **EntryID** is incremented as each report is generated.
    l. When a new GI report is buffered, any older GI report in the buffer is discarded.
    m. The **Resv** attribute of a URCB is now used to reserve the URCB for one client. If a client writes a value of 1 to **Resv**, the server will not allow other clients to write members of the URCB until the first client writes a value of 0. If **Resv**=0, and a client enables the URCB, the server automatically sets **Resv**=1.
    n. 'A pointer to **`MVL_TYPE_CTRL`** is saved in the new **`type_ctrl`** member of **`MVL_VAR_ASSOC`**, so that type information is easier to access.
16. Deleted the functions **`u_mvl_get_nvl`** and **`u_mvl_free_nvl`** from **mvl_uca.c**. Instead, **`mvl_vmd_find_nvl`** is used.
    NOTE: these functions are still used in some code dependent on the **`USE_MANUFACTURED_OBJS`** define, but this is never defined in IEC 61850 or UCA applications.
17. Improved the setting of error codes in MMS Write responses to be more accurate.
18. Changed the IEC 61850 GOOSE sample application to be configured using SCL.

19. Applications using SCL must now include a callback function **u_mvl_scl_set_initial_value** (see **scl_srvr.c**). This function is only called if the SCL parser cannot convert the initial value according to the standard. The user may implement a special conversion in this function, but usually it should just return an error so that parsing stops.

20. Simplified the ODF file for the **scl_srvr** sample application, so that all MMS objects created from SCL.

21. Added the **mvl61850_datset_wr_ind** function to **leafmap.xml** so that it is included in the leaf function pointer table. This is ***important*** because the SCL parsing code must automatically map this function.

22. **IMPORTANT**: Data changes for IEC 61850 Reports are no longer automatically detected by the MVL library. The change detection was flawed because it treated Quality changes as Data changes, and detected changes in irrelevant attributes. Instead, it is now required that the "leaf" functions detect changes, and set the appropriate reason by setting the **rpt_reason** member of **MVL_VAR_ASSOC** (see example in **set_rpt_reason** in **userleaf.c**).

23. Changed **mvlu_find_comp_type** to find 'non-dynamic' types as well as 'dynamic' types. Previously, the function would only find 'dynamic' types.

24. Added the **reserved_1** member to the **RUNTIME_CTRL** structure. It allows the flexibility to store additional type information, and it is very useful during SCL processing.

25. Added several new functions (**ms_rt_bld_***) to construct a type definition without TDL.

26. Added functions to create and maintain multiple VMDs to make client applications easier to implement (the old functions assumed there was only one global VMD). Several functions were changed to take an additional VMD argument and some were replaced with new functions (see *Changes to Support more than one VMD* in a later section details).

27. Deleted the **MVL_DYN_ASN1_TYPES** define and all code that depended on it. This define enabled code to dynamically generate the ASN.1 encoding of type definitions for GetVariableAccessAttributes responses, instead of using static encoding generated by Foundry. From now on, only the dynamic code is provided.

28. Added the functions **mvlu_rpt_ctrl_destroy_all** and **mvl61850_rpt_ctrl_destroy_all**.

29. Changed IEC 61850 control handling so that when an **Oper** is being written, the write is only allowed if the entire **Oper** structure is written in a single MMS Write request. You cannot write a higher level structure that contains an **Oper** structure and you cannot write individual elements of an **Oper** structure.

30. Deleted the global variables **_mvl_curr_net_info** and **_mvl_curr_usr_ind_ctrl**.

31. Completely changed arguments for **scl2_datatype_create_all** and **scl2_ld_create_all**, so that multiple SCL files may be processed and multiple VMDs created.

32. Improved the code that generates type definitions from SCL so that it now properly saves initial values specified in <Val> elements inside <DA> or <BDA> elements. The initial values are written to any variable referencing the type definition. To implement this change, the code was mostly rewritten to use the new **ms_rt_bld_*** functions.

33. Added test code in **client.c** to configure using SCL. The code is triggered by a new command line argument.

34. Reduced the number of different versions of each sample application, built with different versions of the "stack" library.

35. **FILE_LOG_EN** is no longer a default option in **sLogCtrl**. Applications using SISCO logging need to turn this flag **ON** if logging to a log file is desired:

    ```
    sLogCtrl->logCtrl |= LOG_FILE_EN;
    ```

36. MMS_LITE samples were revised to use IPC Logging. The source file slogipc.c was added to the vcproj and makefiles for all sample applications. Please refer to **logcfgx.c**, **client.c**, **server.c**, **uca_srvr.c**, and **scl_srvr.c** for additions/changes to the sample source code.

37. Removed **logcfg.xsd** and **logcfg.dtd** from this product distribution.

38. Changed the IEC 61850 server code to allow the **stVal** attribute of the **Mod** attribute (INC class) or **Beh** attribute (INS class) to be configured in SCL as "Enum" or "INT32". Because the Tissues regarding **CtxInt** (120, 146, 171, and 234) are not entirely clear at this date, this change should allow users to configure whatever type is finally clarified in the standard.

39. The define **ALLOW_MULTIPLE_REQUESTS_OUT**, documented in the Reference Manual, is no longer used. It was deleted from the source code sometime before V4.2951, but the change may not have been documented in earlier releases.

40. Changed the functions in **gensock2.c**, **tp0_socks.c**, and **slogipc.c** to use only one mutex semaphore to avoid potential deadlocks. The semaphore is controlled by the macros **S_LOCK_UTIL_RESOURCES** and **S_UNLOCK_UTIL_RESOURCES** (these macros replace **S_LOCK_RESOURCES** and **S_UNLOCK_RESOURCES**).

41. Changed the "gensock2" interface (in **gensock2.c**) as follows:

    a. Added the capability to manage multiple "gensock" contexts by passing a pointer to a **GEN_SOCK_CTXT** structure to most functions.

    b. Replaced the **uSockConnect** function pointer with 2 separate pointers **uSockConnectInd** (called only when a connect "Indication" is received) and **uSockConnectConf** (called only when a connect "Confirm" is received).

    c. Added the optional functions **sockEventPut** and **sockEventGet**. The **sockEventPut** function puts a socket event on a list. It may be called from a callback function to save an event for later processing. The **sockEventGet** function gets a socket event from the list when the application is ready to process it. These 2 functions are thread-safe.

    d. Removed the **gs_poll_mode** flag that controlled an undocumented and confusing gensock option.

42. Changed the function **u_ml_get_rt_type** in the **mmsl** library to just log an error and return **SD_FAILURE**. The library contains a "default" function which may be replaced by a user-customized function. In the past, the "default" function called "mvl" functions, so it required linking to the **mvl** library, even though most applications never executed this function. Any user that depends on the old behavior must create their own "customized" function which will override the library function (see example in **uca_srvr.c**).

43. New files:

cmd\gnu\gse_mgmt_test.mak	Builds IEC 61850 GSE Mgmt test executable for LINUX, etc.
cmd\gnu\gse_mgmt.mak	Builds IEC 61850 GSE Mgmt library for LINUX, etc.
cmd\gnu\scl_tpxs0.mak	Builds IEC 61850 Sampled Value test executable for LINUX, etc.
cmd\gnu\smpval.mak	Builds IEC 61850 Sampled Value library for LINUX, etc.
cmd\gnu\ositpxs.mak	Builds new stack library for LINUX, etc.
cmd\gnu\cositpxs0.mak	Builds new sample client including IEC 61850 Sampled Value test code.
cmd\gnu\sositpxs0.mak	Builds sample server with ositpxs stack library.
cmd\gnu\uositpxs0.mak	Builds sample UCA server with ositpxs stack library.
cmd\gnu\scl_tpxs0.mak	Builds new sample SCL server including IEC 61850 Sampled Value test code.
cmd\win32\gse_mgmt_test.vcproj	Builds IEC 61850 GSE Mgmt test executable for Windows.
cmd\win32\gse_mgmt.vcproj	Builds IEC 61850 GSE Mgmt library for Windows.
cmd\win32\scl_tpxs0.vcproj	Builds IEC 61850 Sampled Value test executable for Windows.
cmd\win32\smpval.vcproj	Builds IEC 61850 Sampled Value library for Windows.
inc\gse_mgmt.h	Prototypes, etc for IEC 61850 GSE Management support.
mvl\src\mvl61850_rpt.c	Most IEC 61850 reporting code moved from **mvlu_rpt.c** to this file.
mvl\usr\gse_mgmt\*.*	IEC 61850 GSE Management sample application.
src\smpval\smpval_dec.c	Functions to decode IEC 61850 Sampled Value messages.
src\smpval\smpval_enc.c	Functions to encode IEC 61850 Sampled Value messages.
uca\goose\gse_mgmt_dec.c	Functions to decode IEC 61850 GSE Management messages.
uca\goose\gse_mgmt_enc.c	Functions to decode IEC 61850 GSE Management messages.

Deleted file:

clnp_dos.c	Old subnet interface code – no longer used.

MMS Lite (V5.0602) contained the following corrections:

- The handling of MMS Alternate Access on arrays in the **mvlu** library was corrected. The data pointer passed to each "leaf" function (i.e., primData) was changed to point to the correct data.

MMS *Lite* (V5.06) contained the following corrections:

- An assertion was removed from the transaction queuing algorithm in **ethsub.c**.
- The spelling of the IEC 61850 "BufTm" attribute was corrected from "BufTim" when processing SCL.
- Fixed the handling of the "Specification With Result" parameter when processing a MMS Read indication (please refer to **`spec_in_result`** handling in the **`mvlas_read_resp`** function in **s_read.c**).
- The code generated by Foundry for Domain-specific Journals was corrected.
- Fixed buffer overflow problems in the SCL parser when reading attribute values longer than 64 characters. Also eliminated unnecessary copying of SCL data.
- The initial value of DatSet generated from SCL for GOOSE and LOG control blocks was corrected.
- Fixed a memory leak in the **`clnp_read`** function.
- The **`LastApplError`** variable is no longer created at startup but created temporarily only when needed. This prevents clients from seeing the variable in a GetNameList response.
- In **iec_rx.c**, a call to **`clnp_snet_free`** was added to fix a memory leak. Any user application based on this code might need a similar change.
- The Foundry does NOT create **`mvl_rt_tables`** if it would be empty (empty array is illegal).
- Corrected the DeleteNamedVariableList indication processing so that if the "Scope of Delete" is "SPECIFIC", the deletable flag is checked for each NVL before allowing the deletion.
- Changed the type of the "T" attribute of controls from "EntryTime" to "Timestamp", to resolve Tissue #35. Changed the sample SCL file and **iec_cdc.odf**.
- Changed the CMV attribute from "mcval" (incorrect) to "cVal" (correct) in the sample SCL file.
- Fixed the error class and code sent in Error responses for GetVariableAccessAttributes and GetNamedVariableListAttributes indications.
- The length of USERST was corrected in **gentypes.odf**.
- Corrected memory leaks on error conditions in **mvlu_rt.c** and **mvl_typ2.c**.
- The IEC 61850 Controls code was fixed to send a LastApplError report if a write to **`Oper`**, **`SBOw`**, or **`Cancel`** fails. This code depends on the user "leaf" functions to set the report data by filling in the LastApplError member of the **`MVLAS_WR_VA_CTRL`** structure. To see how this should be done, look for LastApplError in **userleaf.c**.
- Corrected a memory leak in **mvl_obj.c** module for the **`last_data`** field.
- Corrected a memory leak in **`u_a_associate_ind`** (**mvl_acse.c**) when the **`u_mvl_connect_ind_ex`** function returned a failure.
- Corrected an ASN.1 decoding problem with ASN.1 Generalized Time values near "1970-01-01 00:00:00" in timezones east of GMT. The problem often occurred when receiving a MMS File Directory response from a server that had uninitialized file timestamps.
- Fixed the handling of **`sboTimeout`** for IEC 61850 controls to use milliseconds (not seconds).
- Corrected the initial value of **`LogRef`** for IEC 61850 Logs to be an ObjectReference.