SatView™ Database Driver Development

All information is subject to change without notice and does not represent a commitment on the part of **BINARY SPACE**.
Release 1.02 (July 2015)

Table of Contents

Appendix

Table of Figures

Document Change Log

Issue	Revision	Date	Affected	Reason for change
1	0	March 2011	All	New document
1	1	June 2011	3.1.2.7.	DBMS Tables Exposure
1	2	July 2015	All	Extended database interface (SCOS-2000 compatibility)

1. Introduction

SatView™ exposes important interfaces to third-party developers which allow an effective customization of the product.

The following areas can be subject of such an extension:

- **Data I/O**
  Writing data I/O device drivers can make SatView™ compatible to any available TM/TC front-end equipment.
- **Database**
  Multiple database standards can be supported by developing database drivers which make a migration of SatView™ to new missions possible in a flexible way.
- **Automation**
  Offers an interface to all automation services of SatView™; ideal for the integration into complex and highly automated ground segments.

This document focuses on the development of database drivers.

2. Interface Architecture

By exposing important interfaces to third-party developers customers can adapt SatView™ to their current environment in a flexible way:
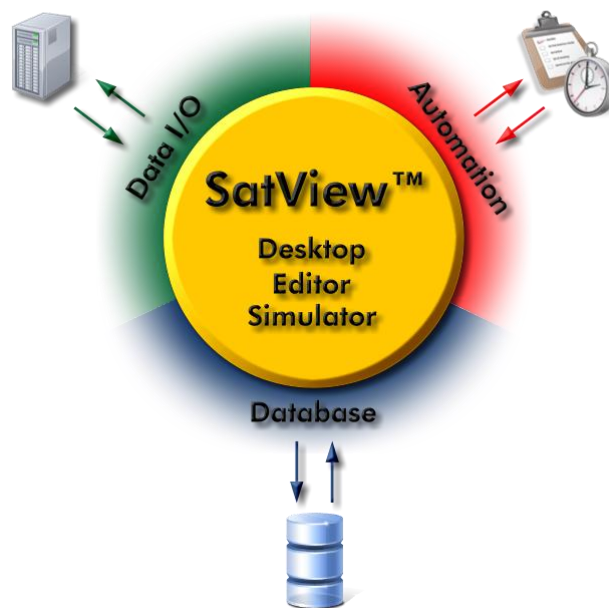

Figure 2.1. - SatView™ Interfaces

The work includes the development of an interface driver in form of a *Dynamic Link Library* (DLL) for the Data I/O and the database; the automation is performed though XML-formatted requests via TCP/IP.

3. Abstract Interface Classes

The interface specification is implemented via C++ abstract base classes exposing the interface through virtual member functions. This approach has the advantage that SatView™ does not have to care about any database specifics because they are encapsulated by the DLL.

3.1. The `CDatabaseEngine` Abstract Base Class

Any database driver to be written for SatView™ must be derived from the base class called `CDatabaseEngine`.

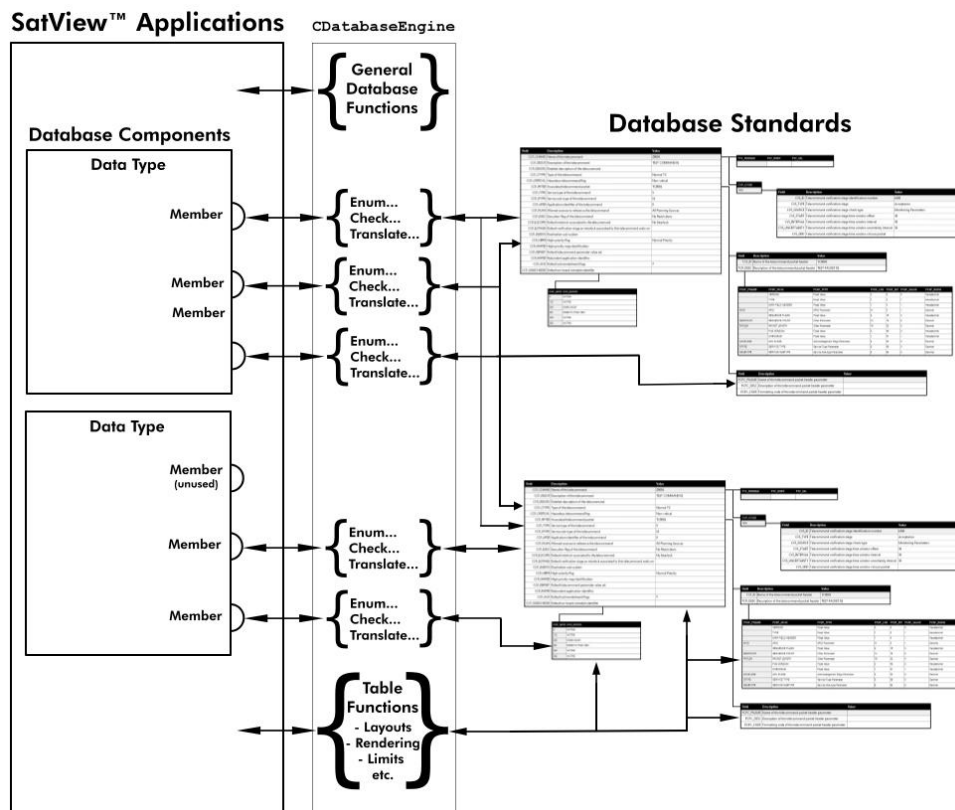The following figure shows the general database driver architecture:

Figure 3.1. - SatView™ Database Driver Architecture

This chapter hereinafter explains the various member functions, types and flags involved with this class.

3.1.1. Non-virtual Functions
The functions listed here do not need to be implemented in any derived class. They provide the basic functionality of the database driver.

```
BOOL Open(CONST CDatabaseProfileInfo *pDatabaseProfileInfo,
         BOOL bLoad=TRUE,
         BOOL bBinary=TRUE)
BOOL Open(CONST CDatabaseProfileInfo *pDatabaseProfileInfo,
         LPCTSTR pszDBMS,
         BOOL bLoad=TRUE,
         BOOL bBinary=TRUE)
```

Opens the database driver.

Parameters:
pDatabaseProfileInfo
> Specifies the database (location, characteristics etc.) to be opened.

pszDBMS
> Specifies the database access token required by the *Database Management System* (DBMS) in order to open (load) the database.

bLoad
> Indicates if the database should be opened and loaded or opened only.

bBinary
> Indicates the type of database to be opened (loaded).
> Specifying TRUE opens the binary (compiled) database file, FALSE opens the DBMS hosted database.

Return Value:
Returns TRUE if the database driver could be opened; FALSE if not.

```
BOOL Initialize(DATABASEMESSAGESBYIDPROC pMessagesIDProc,
         DATABASEMESSAGESBYTEXTPROC pMessagesTextProc,
         DATABASEHELPBYTOPICPROC pHelpTopicProc)
```

Initializes the database driver and installs a callback procedure for messages.

Parameters:

pMessagesIDProc

> Specifies a pointer to a callback procedure provided to display event identifier-based messages issued by the database driver. The supplied procedure must conform to the following syntax:
>
> ```
> BOOL (CALLBACK *)(CONST EVENT_DESCRIPTOR nMessageID);
> ```

pMessagesTextProc

> Specifies a pointer to a callback procedure provided to display text-based messages issued by the database driver. The supplied procedure must conform to the following syntax:
>
> ```
> BOOL (CALLBACK *)(LPCTSTR pszMessage,BOOL bAudition);
> ```

pHelpTopicProc

> Specifies a pointer to a callback procedure provided to open the online help showing the specified topic. The supplied procedure must conform to the following syntax:
>
> ```
> BOOL (CALLBACK *)(LPCTSTR pszTopic);
> ```

Return Value:

Returns TRUE if the initialization of the database driver was successful; FALSE if not.

```
BOOL IsOpen() CONST
```

Opens the database driver.

Return Value:

Returns TRUE if the database driver is currently open; FALSE if not.

```
VOID Close()
```

Closes the database driver.

Return Value:

None.

```
BOOL OpenDBMS(DWORD dwOptions)
```

Opens the DBMS hosted database via the database driver.

Parameters:

dwOptions

Specifies how the database should be opened:

Policy	Description
CDatabase::openExclusive	A data source is always opened as exclusive.
CDatabase::openReadOnly	The database is opened in a read-only mode.
CDatabase::noOdbcDialog	Prevents the display of the ODBC connection dialog box, regardless of whether enough connection information is supplied.
CDatabase::forceOdbcDialog	Always display the ODBC connection dialog box.

Return Value:

Returns TRUE if the database driver could open the DBMS hosted database; FALSE if not.

```
BOOL CloseDBMS()
```

Closes the DBMS hosted database via the database driver.

Return Value:

Returns TRUE if the database driver could close the DBMS hosted database; FALSE if not.

```
CString GetName() CONST
```

Returns the name of the database currently open via the database driver.

Return Value:

Returns the name of the database as a string.

3.1.2. Virtual Functions

The virtual functions make up the actual interface between a database and SatView™. Most of them have to be implemented.

3.1.2.1. Database Manipulation

```
virtual BOOL Save()
virtual BOOL Save(ULONGLONG nComponents)
```

Saves the complete database or components of it to the binary (compiled) database.

Parameters:

nComponents

Specifies which components of a database should be saved:
- DATABASE_COMPONENT_TMPACKET
- DATABASE_COMPONENT_TCFUNCTION
- DATABASE_COMPONENT_TCSEQUENCE
- DATABASE_COMPONENT_TCPROCEDURE
- DATABASE_COMPONENT_OBPROCESSOR
- DATABASE_COMPONENT_TMPARAMETER
- DATABASE_COMPONENT_TCPARAMETER
- DATABASE_COMPONENT_AND
- DATABASE_COMPONENT_GRD
- DATABASE_COMPONENT_MMD
- DATABASE_COMPONENT_POD
- DATABASE_COMPONENT_SCRIPT
- DATABASE_COMPONENTS_ALL

Note:

The above flags can be combined in order to save multiple components at once. For convenience, DATABASE_COMPONENTS_ALL is defined as a combination of all the components.

For some of the above database components it is possible to specify sub-components as follows:

DATABASE_COMPONENT_TMPACKET:
- DATABASE_SUBCOMPONENT_TMPACKETIDENTIFICATION
- DATABASE_SUBCOMPONENT_TMPACKETIDENTCRITERION
- DATABASE_SUBCOMPONENT_TMPACKETHEADER
- DATABASE_SUBCOMPONENT_TMPACKETSTREAM
- DATABASE_SUBCOMPONENT_TMPACKETGROUP

DATABASE_COMPONENT_TCFUNCTION:
- DATABASE_SUBCOMPONENT_TCFUNCTIONPACKET
- DATABASE_SUBCOMPONENT_TCFUNCTIONPACKETPARAMETER
- DATABASE_SUBCOMPONENT_TCFUNCTIONPREEXEGROUP
- DATABASE_SUBCOMPONENT_TCFUNCTIONEXEVERSTAGE
- DATABASE_SUBCOMPONENT_TCFUNCTIONEXEVERGROUP
- DATABASE_SUBCOMPONENT_TCFUNCTIONPARAMETERSET
- DATABASE_SUBCOMPONENT_TCFUNCTIONROUTINGPATH

DATABASE_COMPONENT_TCSEQUENCE:
- DATABASE_SUBCOMPONENT_TCFUNCTIONBLOCK

DATABASE_COMPONENT_OBPROCESSOR:
- DATABASE_SUBCOMPONENT_OBPROCESSORMEMORYIMAGE

DATABASE_COMPONENT_TMPARAMETER:
- DATABASE_SUBCOMPONENT_TMPARAMETERGROUP
- DATABASE_SUBCOMPONENT_TMNUMCALTABLE
- DATABASE_SUBCOMPONENT_TMTXTCALTABLE
- DATABASE_SUBCOMPONENT_TMNUMOOLTABLE
- DATABASE_SUBCOMPONENT_TMTXTOOLTABLE

DATABASE_COMPONENT_TCPARAMETER:
- DATABASE_SUBCOMPONENT_TCNUMCALTABLE
- DATABASE_SUBCOMPONENT_TCTXTCALTABLE
- DATABASE_SUBCOMPONENT_TCNUMOOLTABLE
- DATABASE_SUBCOMPONENT_TCTXTOOLTABLE

DATABASE_COMPONENT_AND:
- DATABASE_SUBCOMPONENT_AND_NORMAL
- DATABASE_SUBCOMPONENT_AND_GENERIC

DATABASE_COMPONENTS_ALL:
- DATABASE_SUBCOMPONENTS_ALL

Note:
- For convenience, DATABASE_SUBCOMPONENTS_ALL is defined as a combination of all the sub-components.
- Depending on the database standard implemented not all subcomponents might be in use.

Return Value:

Returns TRUE if the database driver could save the complete database (or components of it) successfully; FALSE if not.

```
virtual BOOL Save(CONST CDatabaseVersion *pDatabaseVersion)
virtual BOOL Save(CONST CDatabaseTMPacket *pDatabaseTMPacket)
virtual BOOL Save(CONST CDatabaseTCPacket *pDatabaseTCPacket)
virtual BOOL Save(CONST CDatabaseTCFunction *pDatabaseTCFunction)
virtual BOOL Save(CONST CDatabaseTCSequence *pDatabaseTCSequence)
virtual BOOL Save(CONST CDatabaseOBProcessor
          *pDatabaseOBProcessor)
virtual INT Save(CONST CDatabaseTCProcedure *pDatabaseTCProcedure)
virtual BOOL Save(CONST CDatabaseTMParameter
          *pDatabaseTMParameter)
virtual BOOL Save(CONST CDatabaseTCParameter
          *pDatabaseTCParameter)
virtual BOOL Save(CONST CDatabaseTMNumCalTable
          *pDatabaseTMNumCalTable)
virtual BOOL Save(CONST CDatabaseTMTxtCalTable
          *pDatabaseTMTxtCalTable)
virtual BOOL Save(CONST CDatabaseTMNumOolTable
          *pDatabaseTMNumOolTable)
virtual BOOL Save(CONST CDatabaseTMTxtOolTable
          *pDatabaseTMTxtOolTable)
virtual BOOL Save(CONST CDatabaseTCNumCalTable
          *pDatabaseTCNumCalTable)
virtual BOOL Save(CONST CDatabaseTCTxtCalTable
          *pDatabaseTCTxtCalTable)
virtual BOOL Save(CONST CDatabaseTCNumOolTable
          *pDatabaseTCNumOolTable)
virtual BOOL Save(CONST CDatabaseTCTxtOolTable
          *pDatabaseTCTxtOolTable)
virtual BOOL Save(CONST CDatabaseTCPreExeGroup
          *pDatabaseTCPreExeGroup)
virtual BOOL Save(CONST CDatabaseTCExeVerGroup
          *pDatabaseTCExeVerGroup)
virtual BOOL Save(CONST CDatabaseTMPacketGroup
          *pDatabaseTMPacketGroup)
```

```
virtual BOOL Save(CONST CDatabaseTMPacketHeader
        *pDatabaseTMPacketHeader)
virtual BOOL Save(CONST CDatabaseTMPacketStream
        *pDatabaseTMPacketStream)
virtual BOOL Save(CONST CDatabaseTCParameterSet
        *pDatabaseTCParameterSet)
virtual BOOL Save(CONST CDatabaseTCFunctionBlock
        *pDatabaseTCFunctionBlock)
virtual BOOL Save(CONST CDatabaseTMParameterGroup
        *pDatabaseTMParameterGroup)
virtual BOOL Save(CONST CDatabaseTCPacketParameter
        *pDatabaseTCPacketParameter)
virtual BOOL Save(CONST CDatabaseTMPacketIdentification
        *pDatabaseTMPacketIdentification)
virtual BOOL Save(CONST CDatabaseTMPacketIdentificationCriterion
        *pDatabaseTMPacketIdentificationCriterion)
virtual BOOL Save(CONST CDatabaseOBProcessorMemoryImage
        *pDatabaseOBProcessorMemoryImage)
virtual BOOL Save(CONST CANDLayout *pANDLayout)
virtual BOOL Save(CONST CGRDLayout *pGRDLayout)
virtual BOOL Save(CONST CPODLayout *pPODLayout)
```

Saves the supplied database component to the DBMS hosted database.

Parameters:

  pDatabaseVersion
        Specifies the database version information to be saved.
  pDatabaseTMPacket
        Specifies the telemetry packet to be saved.
  pDatabaseTCPacket
        Specifies the telecommand packet to be saved.
  pDatabaseTCFunction
        Specifies the telecommand function to be saved.
  pDatabaseTCSequence
        Specifies the telecommand sequence to be saved.
  pDatabaseOBProcessor
        Specifies the on-board processor to be saved.
  pDatabaseTCProcedure
        Specifies the telecommand procedure to be saved.

pDatabaseTMParameter
Specifies the telemetry parameter to be saved.
pDatabaseTCParameter
Specifies the telecommand parameter to be saved.
pDatabaseTMNumCalTable
Specifies the telemetry related numerical calibration table to be saved.
pDatabaseTMTxtCalTable
Specifies the telemetry related textual calibration table to be saved.
pDatabaseTMNumOolTable
Specifies the telemetry related numerical out-of-limit table to be saved.
pDatabaseTMTxtOolTable
Specifies the telemetry related textual out-of-limit table to be saved.
pDatabaseTCNumCalTable
Specifies the telecommand related numerical calibration table to be saved.
pDatabaseTCTxtCalTable
Specifies the telecommand related textual calibration table to be saved.
pDatabaseTCNumOolTable
Specifies the telecommand related numerical out-of-limit table to be saved.
pDatabaseTCTxtOolTable
Specifies the telecommand related textual out-of-limit table to be saved.
pDatabaseTCPreExeGroup
Specifies the telecommand pre-execution parameter group to be saved.
pDatabaseTCExeVerGroup
Specifies the telecommand execution verification parameter group to be saved.
pDatabaseTMPacketGroup
Specifies the telemetry packet group to be saved.
pDatabaseTMPacketHeader
Specifies the telemetry packet header to be saved.
pDatabaseTMPacketStream
Specifies the telemetry packet stream to be saved.
pDatabaseTCParameterSet
Specifies the telecommand parameter set to be saved.
pDatabaseTCFunctionBlock
Specifies the telecommand function block to be saved.
pDatabaseTMParameterGroup
Specifies the telemetry parameter group to be saved.

pDatabaseTCPacketParameter
> Specifies the telecommand packet parameter group to be saved.

pDatabaseTMPacketIdentification
> Specifies the telemetry packet identification information to be saved.

pDatabaseTMPacketIdentificationCriterion
> Specifies the telemetry packet identification criterion information to be saved.

pDatabaseOBProcessorMemoryImage
> Specifies the on-board memory image to be saved.

pANDLayout
> Specifies the alphanumeric display to be saved.

pGRDLayout
> Specifies the graphic display to be saved.

pPODLayout
> Specifies the parameter observation display to be saved.

Return Value:

Returns TRUE if the database driver could successfully save the database component specified; FALSE if not. In case of *pDatabaseTCProcedure* it returns the index of the saved telecommand procedure.

📝 Note:

Depending on the database standard implemented not all of the above procedures need to be implemented.

```
virtual BOOL Load()
virtual UINT Load(ULONGLONG nComponents,BOOL bAll=TRUE)
```

Loads the complete database or components of it from the binary (compiled) database.

Parameters:

nComponents
> See the 'BOOL **Save**(…)' procedure for more information about this argument.

bAll
> Indicates if the directories of the displays should be loaded too.

Return Value:

Returns TRUE if the database driver could load the complete database (or components of it) successfully; FALSE if not. The value -1 is returned when a new instance of the database was loaded.

```
virtual BOOL Flush(CONST CDatabaseVersions *pDatabaseVersions,BOOL
        bDelete=FALSE)
virtual BOOL Flush(CONST CDatabaseOBProcessor
        *pDatabaseOBProcessor,
        CONST CDatabaseOBProcessorMemoryImage
        *pDatabaseOBProcessorMemoryImage,
        BOOL bDelete=FALSE)
```

Flushes (or deletes) the specified database version information or the on-board processor (incl. the associated memory image) to (or from) the DBMS hosted database.

Parameters:

pDatabaseVersion
> Specifies the database version information to be flushed to the database.

pDatabaseOBProcessor
> Specifies the on-board memory processor to be flushed to the database.

pDatabaseOBProcessorMemoryImage
> Specifies the associated memory image to be flushed to the database.

bDelete
> Indicates if the specified database version information or the on-board memory processor (incl. memory image) should be saved or deleted.

Return Value:

Returns TRUE if the database driver could save (or delete) the specified database components successfully; FALSE if not.

Note:

Depending on the database standard implemented not all of the above procedures need to be implemented.

```
virtual BOOL Clean(CONST CDatabaseTMPacketIdentifications
        *pDatabaseTMPacketIdentifications)
```

Deletes all parameters belonging to one of the telemetry packets identified by the specified collection from the DBMS hosted database.

Parameters:

pDatabaseTMPacketIdentifications
> Specifies a collection of telemetry packet identifications those related parameters should be deleted.
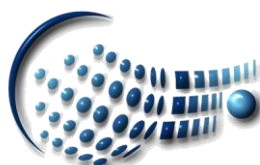
Return Value:

Returns TRUE if the database driver could delete the specified database components successfully; FALSE if not.

 Note:

Depending on the database standard implemented the above procedure does not need to be implemented.

```
virtual BOOL Delete(CONST CDatabaseVersion *pDatabaseVersion)
virtual BOOL Delete(CONST CDatabaseTMPacket *pDatabaseTMPacket)
virtual BOOL Delete(CONST CDatabaseTCPacket *pDatabaseTCPacket)
virtual BOOL Delete(CONST CDatabaseTCFunction
          *pDatabaseTCFunction)
virtual BOOL Delete(CONST CDatabaseTCSequence
          *pDatabaseTCSequence)
virtual BOOL Delete(CONST CDatabaseOBProcessor
          *pDatabaseOBProcessor)
virtual INT Delete(CONST CDatabaseTCProcedure
          *pDatabaseTCProcedure)
virtual BOOL Delete(CONST CDatabaseTMParameter
          *pDatabaseTMParameter)
virtual BOOL Delete(CONST CDatabaseTCParameter
          *pDatabaseTCParameter)
virtual BOOL Delete(CONST CDatabaseTMNumCalTable
          *pDatabaseTMNumCalTable)
virtual BOOL Delete(CONST CDatabaseTMTxtCalTable
          *pDatabaseTMTxtCalTable)
virtual BOOL Delete(CONST CDatabaseTMNumOolTable
          *pDatabaseTMNumOolTable)
virtual BOOL Delete(CONST CDatabaseTMTxtOolTable
          *pDatabaseTMTxtOolTable)
virtual BOOL Delete(CONST CDatabaseTCNumCalTable
          *pDatabaseTCNumCalTable)
virtual BOOL Delete(CONST CDatabaseTCTxtCalTable
          *pDatabaseTCTxtCalTable)
virtual BOOL Delete(CONST CDatabaseTCNumOolTable
          *pDatabaseTCNumOolTable)
```

```
virtual BOOL Delete(CONST CDatabaseTCTxtOolTable
        *pDatabaseTCTxtOolTable)
virtual BOOL Delete(CONST CDatabaseTCPreExeGroup
        *pDatabaseTCPreExeGroup)
virtual BOOL Delete(CONST CDatabaseTCExeVerGroup
        *pDatabaseTCExeVerGroup)
virtual BOOL Delete(CONST CDatabaseTMPacketGroup
        *pDatabaseTMPacketGroup)
virtual BOOL Delete(CONST CDatabaseTMPacketHeader
        *pDatabaseTMPacketHeader)
virtual BOOL Delete(CONST CDatabaseTMPacketStream
        *pDatabaseTMPacketStream)
virtual BOOL Delete(CONST CDatabaseTCParameterSet
        *pDatabaseTCParameterSet)
virtual BOOL Delete(CONST CDatabaseTCFunctionBlock
        *pDatabaseTCFunctionBlock)
virtual BOOL Delete(CONST CDatabaseTMParameterGroup
        *pDatabaseTMParameterGroup)
virtual BOOL Delete(CONST CDatabaseTCPacketParameter
        *pDatabaseTCPacketParameter)
virtual BOOL Delete(CONST CDatabaseTMPacketIdentification
        *pDatabaseTMPacketIdentification)
virtual BOOL Delete(CONST CDatabaseTMPacketIdentificationCriterion
        *pDatabaseTMPacketIdentificationCriterion)
virtual BOOL Delete(CONST CDatabaseOBProcessorMemoryImage
        *pDatabaseOBProcessorMemoryImage)
virtual BOOL Delete(CONST CANDLayout *pANDLayout)
virtual BOOL Delete(CONST CGRDLayout *pGRDLayout)
virtual BOOL Delete(CONST CPODLayout *pPODLayout)
```

Deletes the specified database component from the DBMS hosted database.

Parameters:

See the previously described 'BOOL **Save**(...)' procedures.

Return Value:

Returns TRUE if the database driver could successfully delete the specified component from the DBMS hosted database; FALSE if not. In case of 'INT **Delete**(CONST CDatabaseTCProcedure *pDatabaseTCProcedure)' it returns the previous directory index of the deleted telecommand procedure.

```
virtual BOOL DeleteTMPacket(LPCTSTR pszTag)
virtual BOOL DeleteTCPacket(LPCTSTR pszTag)
virtual BOOL DeleteTCFunction(LPCTSTR pszName)
virtual BOOL DeleteTCSequence(LPCTSTR pszName)
virtual BOOL DeleteOBProcessor(LPCTSTR pszName)
virtual INT DeleteTCProcedure(LPCTSTR pszName)
virtual BOOL DeleteTMParameter(LPCTSTR pszTag)
virtual BOOL DeleteTMNumCalTable(LPCTSTR pszName)
virtual BOOL DeleteTMTxtCalTable(LPCTSTR pszName)
virtual BOOL DeleteTMNumOolTable(LPCTSTR pszName)
virtual BOOL DeleteTMTxtOolTable(LPCTSTR pszName)
virtual BOOL DeleteTCParameter(LPCTSTR pszTag)
virtual BOOL DeleteTCNumCalTable(LPCTSTR pszName)
virtual BOOL DeleteTCTxtCalTable(LPCTSTR pszName)
virtual BOOL DeleteTCNumOolTable(LPCTSTR pszName)
virtual BOOL DeleteTCTxtOolTable(LPCTSTR pszName)
virtual BOOL DeleteTCPreExeGroup(LPCTSTR pszID)
virtual BOOL DeleteTCExeVerStage(LPCTSTR pszID)
virtual BOOL DeleteTCExeVerGroup(LPCTSTR pszID)
virtual BOOL DeleteTMPacketGroup(LPCTSTR pszName)
virtual BOOL DeleteTMPacketHeader(LPCTSTR pszName)
virtual BOOL DeleteTMPacketStream(LPCTSTR pszName)
virtual BOOL DeleteTCParameterSet(LPCTSTR pszName)
virtual BOOL DeleteTCFunctionBlock(LPCTSTR pszName)
virtual BOOL DeleteTMParameterGroup(LPCTSTR pszName)
virtual BOOL DeleteTCPacketParameter(LPCTSTR pszTag)
virtual BOOL DeleteTCParameterValueSet(LPCTSTR pszName)
virtual BOOL DeleteTCFunctionRoutingPath(INT nID)
virtual BOOL DeleteTMPacketIdentification(LPCTSTR pszName)
virtual BOOL DeleteTMPacketIdentificationCriterion(LPCTSTR
        pszName)
virtual BOOL DeleteOBProcessorMemoryImage(LPCTSTR pszName)
virtual BOOL DeleteAND(LPCTSTR pszName)
virtual BOOL DeleteGRD(LPCTSTR pszName)
virtual BOOL DeletePOD(LPCTSTR pszName)
```

Deletes the specified database component from the DBMS hosted database.

Parameters:

pszTag
pszName
pszID
>   Specifies the tag, name or identification of the database component to be deleted.

Return Value:

Returns TRUE if the database driver could successfully delete the specified component from the DBMS hosted database; FALSE if not. In case of '`INT` **`DeleteTCProcedure(`**`LPCTSTR` *`pszName`***`)`**' it returns the previous directory index of the deleted telecommand procedure.

📝 Note:

- The above procedures perform the same action as the overloaded '`BOOL` **`Delete(`**…**`)`**'.
- Depending on the database standard implemented not all of the above procedures need to be implemented.

```
virtual VOID Reset()
```

Resets the content of the database currently opened by the database driver.

Return Value:

None.

```
virtual BOOL Compare(CONST CDatabaseEngine *pDatabase,
        DATABASECOMPAREMESSAGESPROC pCompareMessagesProc,
        UINT nComponents=DATABASE_COMPONENTS_ALL |
        DATABASE_SUBCOMPONENTS_ALL)
```

Compares the database opened by the database driver with the one represented by *pDatabase*.

Parameters:

pDatabase
>   Specifies the database to compare with.

pCompareMessagesProc
>   Specifies a pointer to a callback procedure that will receive all messages generated during the comparison. The specified procedure must conform to the following syntax:
>   `BOOL (CALLBACK *)(UINT nComponent,LPCTSTR pszName,LPCTSTR pszComparedName,LPCTSTR pszMessage)`

nComponents
>   Specifies the database components to compare.
>   See the 'BOOL **Save(...)**' procedure for all possible argument values.

Return Value:

Returns TRUE if the compared database (or components of it) is equal; FALSE if not.

```
virtual BOOL IsComparing() CONST
```

Checks if the database is currently being compared i.e. if the 'BOOL **Compare(...)**' procedure is still executing.

Return Value:

Returns TRUE if the database driver is currently comparing the open database with another one; FALSE if not.

```
virtual BOOL StopComparing()
```

Instructs the database driver to abort the ongoing database comparison.

Return Value:

Returns TRUE if the database driver could stop the database comparison; FALSE if not.

```
virtual BOOL Compile(CBinaryDatabase *pDatabase,
        DATABASECOMPILEMESSAGESPROC pCompileMessagesProc,
        UINT nComponents=DATABASE_COMPONENTS_ALL |
        DATABASE_SUBCOMPONENTS_ALL)
```

Compiles the database opened by the database driver and saves it in a binary (compiled) format.

Parameters:

pDatabase
>   Specifies the binary database to be created or updated by the compilation.

pCompileMessagesProc
>   Specifies a pointer to a callback procedure that will receive all messages generated during the compilation. The specified procedure must conform to the following syntax:
>
>   BOOL (CALLBACK *)(UINT *nComponent*,LPCTSTR *pszName*,LPCTSTR *pszInfo*,LPCTSTR *pszMessage*,BOOL *bWarning*,BOOL *bTimestamp*,BOOL *bAudition*)

nComponents
>> Specifies the database components to compile.
>> See the 'BOOL **Save(…)**' procedure for all possible argument values.

Return Value:

Returns TRUE if the database could be compiled (regardless if errors were detected); FALSE if not.

```
virtual BOOL IsCompiling() CONST
```

Checks if the database is currently being compiled i.e. if the 'BOOL **Compile(…)**' procedure is still executing.

Return Value:

Returns TRUE if the database driver is currently compiling the open database; FALSE if not.

```
virtual BOOL StopCompiling()
```

Instructs the database driver to abort the ongoing database compilation.

Return Value:

Returns TRUE if the database driver could stop the database compilation; FALSE if not.

```
virtual BOOL LoadLibrary(CStringArray &szErrors)
```

Loads all *Dynamic Link Libraries* (DLL) associated with the database currently opened by the database driver.

Parameters:

szErrors
>> Contains all errors that eventually occurred during the loading process.

Return Value:

Returns TRUE if the database driver could successfully load all necessary DLLs; FALSE if not.

```
virtual HINSTANCE GetLibrary() CONST
```

Returns a handle to the top-level *Dynamic Link Library* (DLL) associated with the database currently opened by the database driver.

Return Value:

Returns a handle to the loaded DLL; NULL if an error occurred.

```
virtual VOID FreeLibrary()
```

Releases all *Dynamic Link Libraries* (DLL) previously loaded by `LoadLibrary` and associated with the database currently opened by the database driver.

Return Value:
None.

3.1.2.2. Database Characterization

```
virtual BOOL IsInitialized() CONST
```

Checks if the database currently opened by the database driver has been initialized.

Return Value:
Returns TRUE if the was initialized correctly; FALSE if not.

```
virtual BOOL IsValid() CONST
```

Checks if the database currently opened by the database driver has a valid content.

Return Value:
Returns TRUE if the database is valid; FALSE if not.

```
virtual BOOL IsLocal() CONST
```

Checks if the database currently opened by the database driver is located on the current workstation.

Return Value:
Returns TRUE if the database is hosted locally; FALSE if not.

3.1.2.3. Database Components Access

```
virtual CDatabaseTMPackets *GetTMPackets() CONST
virtual CDatabaseTCPackets *GetTCPackets() CONST
virtual CDatabaseTCFunctions *GetTCFunctions() CONST
virtual CDatabaseTCSequences *GetTCSequences() CONST
virtual CDatabaseTCProcedures *GetTCProcedures() CONST
virtual CDatabaseOBProcessors *GetOBProcessors() CONST
virtual CDatabaseTMParameters *GetTMParameters() CONST
virtual CDatabaseTCParameters *GetTCParameters() CONST
virtual CDatabaseTMNumCalTables *GetTMNumCalTables() CONST
virtual CDatabaseTMTxtCalTables *GetTMTxtCalTables() CONST
virtual CDatabaseTMNumOolTables *GetTMNumOolTables() CONST
virtual CDatabaseTMTxtOolTables *GetTMTxtOolTables() CONST
```

```
virtual CDatabaseTCNumCalTables *GetTCNumCalTables() CONST
virtual CDatabaseTCTxtCalTables *GetTCTxtCalTables() CONST
virtual CDatabaseTCNumOolTables *GetTCNumOolTables() CONST
virtual CDatabaseTCTxtOolTables *GetTCTxtOolTables() CONST
virtual CDatabaseTCPreExeGroups *GetTCPreExeGroups() CONST
virtual CDatabaseTCPreExeGroups *GetTCPreExeGroups() CONST
virtual CDatabaseTCExeVerGroups *GetTCExeVerGroups() CONST
virtual CDatabaseTMPacketGroups *GetTMPacketGroups() CONST
virtual CDatabaseTMPacketHeaders *GetTMPacketHeaders() CONST
virtual CDatabaseTMPacketStreams *GetTMPacketStreams() CONST
virtual CDatabaseTCParameterSets *GetTCParameterSets() CONST
virtual CDatabaseTCFunctionBlocks *GetTCFunctionBlocks() CONST
virtual CDatabaseTMParameterGroups *GetTMParameterGroups() CONST
virtual CDatabaseTCPacketParameters *GetTCPacketParameters() CONST
virtual CDatabaseTMPacketIdentifications
        *GetTMPacketIdentifications() CONST
virtual CDatabaseTMPacketIdentificationCriteria
        *GetTMPacketIdentificationCriteria() CONST
virtual CDatabaseOBProcessorMemoryImages
        *GetOBProcessorMemoryImages() CONST
virtual CDatabaseTMPacket *GetTMPacketSubTags() CONST
```

Returns a pointer to the requested database component directory of the database currently opened by the database driver.

Parameters:

nSubcomponent

Specifies the display sub-type directory to be returned.

See the 'BOOL **Save(...)**' procedure for more information about the possible arguments values.

Return Value:

Returns a pointer to the requested directory; NULL if an error occurred.

Note:

Depending on the database standard implemented not all of the above procedures need to be implemented.

3.1.2.4. Database Displays

The implementation of the database access procedures is optional because the common behavior is inherited by the base class **CBinaryDatabase**.

```
virtual INT EnumANDs(CANDLayouts &pLayouts,
        UINT nSubcomponent=DATABASE_SUBCOMPONENT_AND_NORMAL,
        BOOL bDirectory=TRUE) CONST
virtual INT EnumGRDs(CGRDLayouts &pLayouts,
        BOOL bDirectory=TRUE) CONST
virtual INT EnumMMDs(CMMDLayouts &pLayouts,
        BOOL bDirectory=TRUE) CONST
virtual INT EnumPODs(CPODLayouts &pLayouts,
        BOOL bDirectory=TRUE) CONST
virtual INT EnumScripts(CAutomationScripts &pScripts,
        BOOL bDirectory=TRUE) CONST
virtual INT EnumProcedures(CDatabaseProcedures &pProcedures) CONST
```

Enumerates all displays available in the database currently opened by the database driver.

Parameters:

pLayouts
pScripts
pProcedures
> Contains all displays available of the corresponding type.

nSubSomponent
> Specifies the display sub-type to be enumerated.
> See the 'BOOL **Save**(...)' procedure for more information about the possible arguments values.

bDirectory
> Indicates if only the display directory should be loaded but not the displays.

Return Value:

Returns the number of displays enumerated; -1 if an error occurred.

```
virtual INT EnumANDs(LPCTSTR pszXmlCode,CANDLayouts &pLayouts)
        CONST
virtual INT EnumGRDs(LPCTSTR pszXmlCode,CGRDLayouts &pLayouts)
        CONST
virtual INT EnumPODs(LPCTSTR pszXmlCode,CPODLayouts &pLayouts)
        CONST
```

Parses a XML-based display layout definition file and enumerates all found displays.

Parameters:

pszXmlCode
>   Specifies the file containing the display layout definitions.

pLayouts
>   Contains all displays available of the corresponding type.

Return Value:

Returns the number of displays enumerated; -1 if an error occurred.

```cpp
virtual INT SaveAND(CONST CANDLayout &cLayout,
         UINT nSubcomponent=DATABASE_SUBCOMPONENT_AND_NORMAL)
virtual INT SaveGRD(CONST CGRDLayout &cLayout)
virtual INT SaveMMD(CONST CMMDLayout &cLayout)
virtual INT SavePOD(CONST CPODLayout &cLayout)
virtual INT SaveScript(CONST CAutomationScript &cScript)
```

Saves the supplied display to the database currently opened by the database driver.

Parameters:

cLayout
cScript
>   Contains the display to be saved to the database.

nSubcomponent
>   Specifies the display sub-type to be saved.
>   See the 'BOOL **Save**(…)' procedure for more information about the possible arguments values.

Return Value:

Returns the directory index of the display saved; -1 if an error occurred.

```cpp
virtual BOOL LoadAND(LPCTSTR pszName,CANDLayout &cLayout,
         UINT nSubcomponent=DATABASE_SUBCOMPONENT_AND_NORMAL)
         CONST
virtual BOOL LoadGRD(LPCTSTR pszName,CGRDLayout &cLayout) CONST
virtual BOOL LoadMMD(LPCTSTR pszName,CMMDLayout &cLayout) CONST
virtual BOOL LoadPOD(LPCTSTR pszName,CPODLayout &cLayout) CONST
virtual BOOL LoadScript(LPCTSTR pszName,CAutomationScript
         &cScript) CONST
```

Loads the requested display from the database currently opened by the database driver.

Parameters:

pszName
> Specifies the name of the display to be loaded.

nSubSomponent
> Specifies the display sub-type to be loaded.
> See the 'BOOL **Save(…)**' procedure for more information about the possible arguments values.

cLayout
cScript
> Contains the requested display loaded from the database.

Return Value:

Returns TRUE if the requested display could be loaded; FALSE if not.

```
virtual INT DeleteAND(CONST CANDLayout &cLayout,
         UINT nSubcomponent=DATABASE_SUBCOMPONENT_AND_NORMAL)
virtual INT DeleteGRD(CONST CGRDLayout &cLayout)
virtual INT DeleteMMD(CONST CMMDLayout &cLayout)
virtual INT DeletePOD(CONST CPODLayout &cLayout)
virtual INT DeleteScript(CONST CAutomationScript &cScript)
```

Deletes the supplied display from the database currently opened by the database driver.

Parameters:

cLayout
cScript
> Contains the display to be deleted from the database.

nSubcomponent
> Specifies the display sub-type to be deleted.
> See the 'BOOL **Save(…)**' procedure for more information about the possible arguments values.

Return Value:

Returns the previous directory index of the display deleted; -1 if an error occurred.

3.1.2.5. Database Parsing

The database parsing procedures perform the database consistency & business rules checks. For each of the database components there are a number of functions to check the various aspects of it.

Because SatView™ supports multiple database standards, all data from any database needs to be converted into common internal data types. This is achieved by an API consisting of functions handling the checking and parsing of each of the data type members. These procedures are expanded by others providing translation services between DBMS related values and their corresponding human-readable text:

Function Prefix	Description
Check…	Functions starting with this prefix perform checking services on the related DBMS table field.
Enum…	This prefix indicates that the corresponding function enumerates all valid values for the related DBMS table field.
Translate…	These functions translate a DBMS table field value into the corresponding human-readable text and vice versa.

📝 Note:

Depending on the database standard implemented not all of the following procedures need to be implemented.

3.1.2.5.1. Versioning

```
virtual BOOL CheckVersionName(LPCTSTR pszName) CONST
virtual INT GetVersionNameLength() CONST
virtual BOOL CheckVersionComment(LPCTSTR pszComment) CONST
virtual INT GetVersionCommentLength() CONST
virtual BOOL CheckVersionDomain(LPCTSTR pszDomain) CONST
virtual INT GetVersionDomainLength() CONST
virtual INT TranslateVersionDomain(LPCTSTR pszDomain) CONST
virtual CString TranslateVersionDomain(INT nDomain) CONST
virtual BOOL CheckVersionRelease(LPCTSTR pszRelease) CONST
virtual INT GetVersionReleaseLength() CONST
virtual INT TranslateVersionRelease(LPCTSTR pszRelease) CONST
virtual CString TranslateVersionRelease(INT nRelease) CONST
virtual BOOL CheckVersionIssue(LPCTSTR pszIssue) CONST
virtual INT GetVersionIssueLength() CONST
virtual INT TranslateVersionIssue(LPCTSTR pszIssue) CONST
virtual CString TranslateVersionIssue(INT nIssue) CONST
```

3.1.2.5.2. Telemetry Packets

```
virtual BOOL CheckTMPacketTag(LPCTSTR pszTag) CONST
virtual INT GetTMPacketTagLength() CONST
virtual CString GetTMPacketSubTag() CONST
virtual CString GetTMPacketDummyTag() CONST
virtual BOOL CheckTMPacketID(UINT nID) CONST
virtual BOOL GetTMPacketIDRange(UINT &nLow,UINT &nHigh) CONST
virtual BOOL CheckTMPacketType(INT nType) CONST
virtual INT GetTMPacketTypeLength() CONST
virtual BOOL GetTMPacketTypeRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMPacketType(LPCTSTR pszType) CONST
virtual CString TranslateTMPacketType(INT nType) CONST
virtual BOOL CheckTMPacketSubType(INT nSubType) CONST
virtual BOOL GetTMPacketSubTypeRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMPacketDescriptionLength() CONST
virtual BOOL CheckTMPacketDetails(LPCTSTR pszDetails) CONST
virtual INT GetTMPacketDetailsLength() CONST
virtual BOOL CheckTMPacketSubSystem(LPCTSTR pszSubSystem) CONST
virtual INT GetTMPacketSubSystemLength() CONST
virtual INT EnumTMPacketDataFieldHeaderFlags(CStringArray &szFlags,CUIntArray
          &nFlags) CONST
virtual INT TranslateTMPacketDataFieldHeaderFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMPacketDataFieldHeaderFlag(INT nFlag,BOOL bDBMS=TRUE)
          CONST
virtual BOOL CheckTMPacketDataFieldHeader(INT bPresent) CONST
virtual BOOL CheckTMPacketDataFieldHeaderName(LPCTSTR pszName) CONST
virtual INT GetTMPacketDataFieldHeaderNameLength() CONST
virtual INT EnumTMPacketAPIDs(CStringArray &szAPIDs,CUIntArray &nAPIDs) CONST
virtual BOOL CheckTMPacketAPID(LPCTSTR pszTag,INT nAPID) CONST
virtual BOOL CheckTMPacketAPID(INT nAPID) CONST
virtual INT GetTMPacketAPIDLength() CONST
virtual BOOL GetTMPacketAPIDRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMPacketAPID(LPCTSTR pszAPID,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketAPID(INT nAPID,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMPacketTMTypes(CStringArray &szTMTypes,CUIntArray &nTMTypes)
          CONST
virtual BOOL CheckTMPacketTMType(INT nTMType) CONST
virtual INT GetTMPacketTMTypeLength() CONST
virtual INT TranslateTMPacketTMType(LPCTSTR pszTMType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketTMType(INT nTMType,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTMPacketTMID(INT nTMID) CONST
virtual INT GetTMPacketTMIDLength() CONST
virtual BOOL GetTMPacketTMIDRange(UINT &nLow,UINT &nHigh) CONST
```

```cpp
virtual INT TranslateTMPacketTMID(LPCTSTR pszTMID,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketTMID(INT nTMID,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTMPacket3Disc(INT n3Disc) CONST
virtual INT GetTMPacket3DiscLength() CONST
virtual BOOL GetTMPacket3DiscRange(UINT &nLow,UINT &nHigh) CONST
virtual INT TranslateTMPacket3Disc(LPCTSTR psz3Disc,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacket3Disc(INT n3Disc,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTMPacketFirstFieldOffset(INT nOffset) CONST
virtual BOOL GetTMPacketFirstFieldOffsetRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketFirstFieldWidth(INT nWidth) CONST
virtual BOOL GetTMPacketFirstFieldWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketFirstFieldValue(INT nValue) CONST
virtual BOOL GetTMPacketFirstFieldValueRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketSecondFieldOffset(INT nOffset) CONST
virtual BOOL GetTMPacketSecondFieldOffsetRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketSecondFieldWidth(INT nWidth) CONST
virtual BOOL GetTMPacketSecondFieldWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketSecondFieldValue(INT nValue) CONST
virtual BOOL GetTMPacketSecondFieldValueRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketDataFieldHeaderSize(INT nSize) CONST
virtual BOOL GetTMPacketDataFieldHeaderSizeRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketContentID(INT nID) CONST
virtual BOOL GetTMPacketContentIDRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketTransmissionRate(INT nType,INT nInterval) CONST
virtual BOOL GetTMPacketTransmissionRateRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketTransmissionRetries(INT nRetries) CONST
virtual INT GetTMPacketTransmissionRetriesLength() CONST
virtual BOOL GetTMPacketTransmissionRetriesRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMPacketTransmissionRetries(LPCTSTR pszRetries) CONST
virtual CString TranslateTMPacketTransmissionRetries(INT nRetries) CONST
virtual INT EnumTMPacketTimestampFlags(CStringArray &szFlags,CUIntArray &nFlags)
        CONST
virtual BOOL CheckTMPacketTimestampFlag(LPCTSTR pszFlag) CONST
virtual INT GetTMPacketTimestampFlagLength() CONST
virtual INT TranslateTMPacketTimestampFlag(LPCTSTR pszFlag,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTMPacketTimestampFlag(INT nFlag,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTMPacketVirtualChannel(INT nAPID,INT nChannelID) CONST
virtual BOOL GetTMPacketVirtualChannelRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTMPacketEventTypes(CStringArray &szTypes,CUIntArray &nTypes)
        CONST
virtual BOOL CheckTMPacketEventType(UINT nType) CONST
virtual INT GetTMPacketEventTypeLength() CONST
virtual INT TranslateTMPacketEventType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketEventType(INT nType,BOOL bDBMS=TRUE) CONST
```

```
virtual BOOL CheckTMPacketEventID(INT nType,LPCTSTR pszEventID) CONST
virtual BOOL CheckTMPacketEventID(LPCTSTR pszEventID) CONST
virtual INT GetTMPacketEventIDLength() CONST
virtual INT EnumTMPacketCyclicRedundancyCheckFlags(CStringArray
        &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckTMPacketCyclicRedundancyCheckFlag(INT nFlag) CONST
virtual INT GetTMPacketCyclicRedundancyCheckFlagLength() CONST
virtual INT TranslateTMPacketCyclicRedundancyCheckFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMPacketCyclicRedundancyCheckFlag(INT nFlag,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTMPacketIdentificationValidityFlags(CStringArray
        &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckTMPacketIdentificationValidityFlag(LPCTSTR pszFlag) CONST
virtual INT GetTMPacketIdentificationValidityFlagLength() CONST
virtual INT TranslateTMPacketIdentificationValidityFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMPacketIdentificationValidityFlag(INT nFlag,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTMPacketDataSizes(CStringArray &szSizes,CUIntArray &nSizes)
        CONST
virtual BOOL CheckTMPacketDataSize(INT cbData,INT nFlag) CONST
virtual BOOL GetTMPacketDataSizeRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMPacketDataSize(LPCTSTR pszSize,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketDataSize(INT nSize,BOOL bDBMS=TRUE) CONST
virtual INT GetTMPacketDataSize() CONST
virtual INT GetTMPacketDataFieldHeaderSize() CONST
virtual INT GetTMPacketTotalSize() CONST
virtual INT EnumTMPacketStatus(CStringArray &szStatus,CUIntArray &nStatus) CONST
virtual BOOL CheckTMPacketStatus(LPCTSTR pszStatus) CONST
virtual INT GetTMPacketStatusLength() CONST
virtual INT TranslateTMPacketStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

Telemetry Packet Headers:

```
virtual BOOL CheckTMPacketHeaderName(LPCTSTR pszName) CONST
virtual INT GetTMPacketHeaderNameLength() CONST
virtual INT EnumTMPacketHeaderTMTypeFlags(CStringArray &szFlags,CUIntArray
        &nFlags) CONST
virtual INT TranslateTMPacketHeaderTMTypeFlag(LPCTSTR pszFlag,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTMPacketHeaderTMTypeFlag(INT nFlag,BOOL bDBMS=TRUE)
        CONST
virtual BOOL CheckTMPacketHeaderTMTypePosition(INT nByte,INT nBit,INT nLength)
        CONST
```

```
virtual BOOL GetTMPacketHeaderTMTypePositionRange(INT &nOffsetLow,INT
         &nOffsetHigh,INT &nLengthLow,INT &nLengthHigh) CONST
virtual INT EnumTMPacketHeaderTMIDFlags(CStringArray &szFlags,CUIntArray
         &nFlags) CONST
virtual INT TranslateTMPacketHeaderTMIDFlag(LPCTSTR pszFlag,BOOL bDBMS=TRUE)
         CONST
virtual CString TranslateTMPacketHeaderTMIDFlag(INT nFlag,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTMPacketHeaderTMIDPosition(INT nByte,INT nBit,INT nLength)
         CONST
virtual BOOL GetTMPacketHeaderTMIDPositionRange(INT &nOffsetLow,INT
         &nOffsetHigh,INT &nLengthLow,INT &nLengthHigh) CONST
virtual INT EnumTMPacketHeader3DiscFlags(CStringArray &szFlags,CUIntArray
         &nFlags) CONST
virtual INT TranslateTMPacketHeader3DiscFlag(LPCTSTR pszFlag,BOOL bDBMS=TRUE)
         CONST
virtual CString TranslateTMPacketHeader3DiscFlag(INT nFlag,BOOL bDBMS=TRUE)
         CONST
virtual BOOL CheckTMPacketHeader3DiscPosition(INT nByte,INT nBit,INT nLength)
         CONST
virtual BOOL GetTMPacketHeader3DiscPositionRange(INT &nOffsetLow,INT
         &nOffsetHigh,INT &nLengthLow,INT &nLengthHigh) CONST
virtual INT EnumTMPacketHeaderTimestampFlags(CStringArray &szFlags,CUIntArray
         &nFlags) CONST
virtual INT TranslateTMPacketHeaderTimestampFlag(LPCTSTR pszFlag,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateTMPacketHeaderTimestampFlag(INT nFlag,BOOL bDBMS=TRUE)
         CONST
virtual BOOL CheckTMPacketHeaderTimestampPosition(INT nByte,INT nBit,INT
         nLength) CONST
virtual BOOL GetTMPacketHeaderTimestampPositionRange(INT &nOffsetLow,INT
         &nOffsetHigh,INT &nLengthLow,INT &nLengthHigh) CONST
```

Telemetry Packet Streams:
```
virtual BOOL CheckTMPacketStreamName(LPCTSTR pszName) CONST
virtual INT GetTMPacketStreamNameLength() CONST
virtual INT EnumTMPacketStreamAPIDs(CStringArray &szAPIDs,CUIntArray &nAPIDs)
         CONST
virtual BOOL CheckTMPacketStreamAPID(INT nAPID) CONST
virtual INT GetTMPacketStreamAPIDLength() CONST
virtual INT TranslateTMPacketStreamAPID(LPCTSTR pszAPID,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketStreamAPID(INT nAPID,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTMPacketStreamTMID(INT nTMID) CONST
virtual INT GetTMPacketStreamTMIDLength() CONST
virtual BOOL GetTMPacketStreamTMIDRange(UINT &nLow,UINT &nHigh) CONST
virtual INT TranslateTMPacketStreamTMID(LPCTSTR pszTMID,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketStreamTMID(INT nTMID,BOOL bDBMS=TRUE) CONST
```

```
virtual BOOL CheckTMPacketStreamPeriod(INT nPeriod) CONST
virtual BOOL GetTMPacketStreamPeriodRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketStreamTimeout(INT nTimeout) CONST
virtual BOOL GetTMPacketStreamTimeoutRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketStreamValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTMPacketStreamValidityParameterLength() CONST
virtual BOOL CheckTMPacketStreamValidityValue(INT nValue) CONST
virtual BOOL GetTMPacketStreamValidityValueRange(INT &nLow,INT &nHigh) CONST
```

Telemetry Packet Parameters:

```
virtual BOOL CheckTMPacketParameterTag(LPCTSTR pszTag) CONST
virtual INT GetTMPacketParameterTagLength() CONST
virtual BOOL CheckTMPacketParameterDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMPacketParameterDescriptionLength() CONST
virtual BOOL CheckTMPacketParameterSequenceNumber(INT nNumber) CONST
virtual BOOL GetTMPacketParameterSequenceNumberRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketParameterGroupSize(INT nSize) CONST
virtual BOOL GetTMPacketParameterGroupSizeRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketParameterGroupRepetitionCount(INT nCount) CONST
virtual BOOL GetTMPacketParameterGroupRepetitionCountRange(INT &nLow,INT &nHigh)
        CONST
virtual INT EnumTMPacketParameterChoiceFlags(CStringArray &szFlags,CUIntArray
        &nFlags) CONST
virtual BOOL CheckTMPacketParameterChoiceFlag(UINT nFlag) CONST
virtual INT GetTMPacketParameterChoiceFlagLength() CONST
virtual UINT TranslateTMPacketParameterChoiceFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMPacketParameterChoiceFlag(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTMPacketParameterOnBoardIDFlags(CStringArray &szFlags,CUIntArray
        &nFlags) CONST
virtual BOOL CheckTMPacketParameterOnBoardIDFlag(UINT nFlag) CONST
virtual INT GetTMPacketParameterOnBoardIDFlagLength() CONST
virtual UINT TranslateTMPacketParameterOnBoardIDFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMPacketParameterOnBoardIDFlag(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckTMPacketParameterBytePosition(INT nByte) CONST
virtual BOOL GetTMPacketParameterBytePositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketParameterBitPosition(INT nBit) CONST
virtual BOOL GetTMPacketParameterBitPositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketParameterWidth(INT nWidth) CONST
virtual BOOL GetTMPacketParameterWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketParameterOffset(INT nOffset) CONST
virtual BOOL GetTMPacketParameterOffsetRange(INT &nLow,INT &nHigh) CONST
```

```
virtual BOOL CheckTMPacketParameterOccurrence(INT nOccurrence) CONST
virtual BOOL GetTMPacketParameterOccurrenceRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMPacketParameterOccurrenceCount(INT nCount) CONST
virtual BOOL GetTMPacketParameterOccurrenceCountRange(INT &nLow,INT &nHigh)
         CONST
virtual BOOL CheckTMPacketParameterOccurrenceInterval(INT nOccurrence,INT
         nInterval) CONST
virtual BOOL GetTMPacketParameterOccurrenceIntervalRange(INT &nLow,INT &nHigh)
         CONST
virtual BOOL CheckTMPacketParameterOccurrenceTimeOffset(INT nOffset) CONST
virtual BOOL GetTMPacketParameterOccurrenceTimeOffsetRange(INT &nLow,INT &nHigh)
         CONST
virtual BOOL CheckTMPacketParameterOccurrenceTimeInterval(INT nOccurrence,INT
         nInterval) CONST
virtual BOOL GetTMPacketParameterOccurrenceTimeIntervalRange(INT &nLow,INT
         &nHigh) CONST
virtual BOOL CheckTMPacketParameterValidity(LPCTSTR pszTag) CONST
virtual INT GetTMPacketParameterValidityLength() CONST
virtual INT EnumTMPacketParameterDisplayColumnsFlags(CStringArray
         &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckTMPacketParameterDisplayColumnsFlag(UINT nFlag) CONST
virtual INT GetTMPacketParameterDisplayColumnsFlagRange(UINT &nLow,UINT &nHigh)
         CONST
virtual UINT TranslateTMPacketParameterDisplayColumnsFlag(LPCTSTR pszFlag,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateTMPacketParameterDisplayColumnsFlag(UINT
         nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMPacketParameterDisplayAlignmentFlags(CStringArray
         &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckTMPacketParameterDisplayAlignmentFlag(UINT nFlag) CONST
virtual INT GetTMPacketParameterDisplayAlignmentFlagLength() CONST
virtual UINT TranslateTMPacketParameterDisplayAlignmentFlag(LPCTSTR pszFlag,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateTMPacketParameterDisplayAlignmentFlag(UINT
         nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMPacketParameterDisplayNewLineFlags(CStringArray
         &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckTMPacketParameterDisplayNewLineFlag(UINT nFlag) CONST
virtual INT GetTMPacketParameterDisplayNewLineFlagLength() CONST
virtual UINT TranslateTMPacketParameterDisplayNewLineFlag(LPCTSTR pszFlag,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateTMPacketParameterDisplayNewLineFlag(UINT
         nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMPacketParameterDisplayFormatFlags(CStringArray
         &szFlags,CUIntArray &nFlags) CONST
```

```
virtual BOOL CheckTMPacketParameterDisplayFormatFlag(UINT nFlag) CONST
virtual INT GetTMPacketParameterDisplayFormatFlagLength() CONST
virtual UINT TranslateTMPacketParameterDisplayFormatFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMPacketParameterDisplayFormatFlag(UINT
        nAttributes,BOOL bDBMS=TRUE) CONST
```
Telemetry Packet Groups:
```
virtual BOOL CheckTMPacketGroupName(LPCTSTR pszName) CONST
virtual INT GetTMPacketGroupNameLength() CONST
virtual BOOL CheckTMPacketGroupDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMPacketGroupDescriptionLength() CONST
virtual BOOL CheckTMPacketGroupType(LPCTSTR pszType) CONST
virtual INT GetTMPacketGroupTypeLength() CONST
virtual INT TranslateTMPacketGroupType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMPacketGroupType(INT nType,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTMPacketGroupMember(UINT nID) CONST
```

3.1.2.5.3. Telecommand Packets

```
virtual BOOL CheckTCPacketTag(LPCTSTR pszTag) CONST
virtual INT GetTCPacketTagLength() CONST
virtual BOOL CheckTCPacketType(INT nType) CONST
virtual INT GetTCPacketTypeLength() CONST
virtual INT TranslateTCPacketType(LPCTSTR pszType) CONST
virtual CString TranslateTCPacketType(INT nType) CONST
virtual BOOL CheckTCPacketDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCPacketDescriptionLength() CONST
virtual BOOL CheckTCPacketDetails(LPCTSTR pszDetails) CONST
virtual INT GetTCPacketDetailsLength() CONST
virtual INT EnumTCPacketDataFieldHeaderFlags(CStringArray &szFlags,CUIntArray
        &nFlags) CONST
virtual INT TranslateTCPacketDataFieldHeaderFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCPacketDataFieldHeaderFlag(INT nFlag,BOOL bDBMS=TRUE)
        CONST
virtual BOOL CheckTCPacketDataFieldHeader(INT bPresent) CONST
virtual INT EnumTCPacketAPIDs(CStringArray &szAPIDs,CUIntArray &nAPIDs) CONST
virtual BOOL CheckTCPacketAPID(INT nAPID) CONST
virtual INT GetTCPacketAPIDLength() CONST
virtual INT TranslateTCPacketAPID(LPCTSTR pszAPID,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCPacketAPID(INT nAPID,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCPacketDataSizes(CStringArray &szSizes,CUIntArray &nSizes)
        CONST
virtual BOOL GetTCPacketDataSizeRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTCPacketDataSize(LPCTSTR pszSize,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCPacketDataSize(INT nSize,BOOL bDBMS=TRUE) CONST
```

```
virtual BOOL CheckTCPacketDataSize(ULONGLONG nAttributes,INT nFlag,INT cbData)
          CONST
virtual INT GetTCPacketDataSize() CONST
virtual INT GetTCPacketDataFieldHeaderSize() CONST
virtual INT GetTCPacketTotalSize() CONST
virtual INT EnumTCPacketStatus(CStringArray &szStatus,CUIntArray &nStatus) CONST
virtual BOOL CheckTCPacketStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCPacketStatusLength() CONST
virtual INT TranslateTCPacketStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCPacketStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

Telecommand Packet Parameters:

```
virtual BOOL CheckTCPacketParameterTag(LPCTSTR pszTag) CONST
virtual INT GetTCPacketParameterTagLength() CONST
virtual BOOL CheckTCPacketParameterDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCPacketParameterDescriptionLength() CONST
virtual INT EnumTCPacketParameterTypes(CStringArray &szTypes,CLongUIntArray
          &nTypes) CONST
virtual BOOL CheckTCPacketParameterType(LPCTSTR pszTag,ULONGLONG nType) CONST
virtual INT GetTCPacketParameterTypeLength() CONST
virtual ULONGLONG TranslateTCPacketParameterType(LPCTSTR pszType,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCPacketParameterType(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCPacketParameterValueInterpretations(CStringArray
          &szInterpretations,CLongUIntArray &nInterpretations) CONST
virtual BOOL CheckTCPacketParameterValueInterpretation(ULONGLONG
          nAttributes,ULONGLONG nInterpretation) CONST
virtual INT GetTCPacketParameterValueInterpretationLength() CONST
virtual ULONGLONG TranslateTCPacketParameterValueInterpretation(LPCTSTR
          pszInterpretation,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCPacketParameterValueInterpretation(ULONGLONG
          nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCPacketParameterValueCodings(CStringArray
          &szCodings,CLongUIntArray &nCodings) CONST
virtual BOOL CheckTCPacketParameterValueCoding(ULONGLONG nAttributes,ULONGLONG
          nCoding) CONST
virtual INT GetTCPacketParameterValueCodingLength() CONST
virtual ULONGLONG TranslateTCPacketParameterValueCoding(LPCTSTR pszCoding,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCPacketParameterValueCoding(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCPacketParameterValueRadixes(CStringArray
          &szRadixes,CLongUIntArray &nRadixes) CONST
virtual BOOL CheckTCPacketParameterValueRadix(ULONGLONG nAttributes,ULONGLONG
          nRadix) CONST
```

```
virtual INT GetTCPacketParameterValueRadixLength() CONST
virtual ULONGLONG TranslateTCPacketParameterValueRadix(LPCTSTR pszRadix,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCPacketParameterValueRadix(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckTCPacketParameterValueBitWidth(INT nWidth) CONST
virtual BOOL GetTCPacketParameterValueBitWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCPacketParameterValueBitOffset(INT nOffset) CONST
virtual BOOL GetTCPacketParameterValueBitOffsetRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCPacketParameterConstValue(ULONGLONG nAttributes,LPCTSTR
        pszConstValue) CONST
virtual INT GetTCPacketParameterConstValueLength() CONST
virtual ULONGLONG TranslateTCPacketParameterConstValue(ULONGLONG nAttributes,INT
        nWidth,LPCTSTR pszConstValue) CONST
virtual CString TranslateTCPacketParameterConstValue(ULONGLONG nAttributes,INT
        nWidth,ULONGLONG nConstValue) CONST
virtual BOOL CheckTCPacketParameterConstTimeType(LPCTSTR pszTag) CONST
virtual BOOL CheckTCPacketParameterConstTime(ULONGLONG nAttributes,LPCTSTR
        pszConstTime) CONST
virtual INT GetTCPacketParameterConstTimeLength() CONST
virtual CTimeTag TranslateTCPacketParameterConstTime(ULONGLONG
        nAttributes,LPCTSTR pszConstTime) CONST
```

3.1.2.5.4. Telecommand Functions

```
virtual BOOL CheckTCFunctionName(LPCTSTR pszName) CONST
virtual INT GetTCFunctionNameLength() CONST
virtual BOOL CheckTCFunctionDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCFunctionDescriptionLength() CONST
virtual BOOL CheckTCFunctionExtendedDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCFunctionExtendedDescriptionLength() CONST
virtual BOOL CheckTCFunctionDetails(LPCTSTR pszDetails) CONST
virtual INT GetTCFunctionDetailsLength() CONST
virtual INT EnumTCFunctionTypes(CStringArray &szTypes,CLongUIntArray &nTypes)
        CONST
virtual BOOL CheckTCFunctionType(ULONGLONG nType) CONST
virtual INT GetTCFunctionTypeLength() CONST
virtual ULONGLONG TranslateTCFunctionType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionType(ULONGLONG nAttributes,BOOL bDBMS=TRUE)
        CONST
virtual BOOL CheckTCFunctionAPID(INT nAPID) CONST
virtual BOOL GetTCFunctionAPIDRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCFunctionTCTypes(CStringArray &szTCTypes,CLongUIntArray
        &nTCTypes) CONST
virtual BOOL CheckTCFunctionTCType(ULONGLONG nTCType) CONST
```

```
virtual BOOL GetTCFunctionTCTypeRange(INT &nLow,INT &nHigh) CONST
virtual INT GetTCFunctionTCTypeLength() CONST
virtual ULONGLONG TranslateTCFunctionTCType(LPCTSTR pszTCType,BOOL bDBMS=TRUE)
          CONST
virtual CString TranslateTCFunctionTCType(ULONGLONG nAttributes,BOOL bDBMS=TRUE)
          CONST
virtual BOOL CheckTCFunctionTCID(INT nTCID) CONST
virtual BOOL GetTCFunctionTCIDRange(INT &nLow,INT &nHigh) CONST
virtual INT GetTCFunctionTCIDLength() CONST
virtual INT TranslateTCFunctionTCID(LPCTSTR pszTCID) CONST
virtual CString TranslateTCFunctionTCID(INT nTCID) CONST
virtual BOOL CheckTCFunctionMapID(INT nMapID) CONST
virtual BOOL GetTCFunctionMapIDRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCFunctionPacketTag(LPCTSTR pszTag) CONST
virtual INT GetTCFunctionPacketTagLength() CONST
virtual INT EnumTCFunctionExecutionFlags(CStringArray &szFlags,CLongUIntArray
          &nFlags) CONST
virtual BOOL CheckTCFunctionExecutionFlag(ULONGLONG nFlag) CONST
virtual INT GetTCFunctionExecutionFlagLength() CONST
virtual ULONGLONG TranslateTCFunctionExecutionFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionExecutionFlag(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCFunctionPriorityFlags(CStringArray &szFlags,CLongUIntArray
          &nFlags) CONST
virtual BOOL CheckTCFunctionPriorityFlag(ULONGLONG nFlag) CONST
virtual INT GetTCFunctionPriorityFlagLength() CONST
virtual ULONGLONG TranslateTCFunctionPriorityFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionPriorityFlag(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCFunctionHazardousFlags(CStringArray &szFlags,CLongUIntArray
          &nFlags) CONST
virtual BOOL CheckTCFunctionHazardousFlag(ULONGLONG nFlag) CONST
virtual INT GetTCFunctionHazardousFlagLength() CONST
virtual ULONGLONG TranslateTCFunctionHazardousFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionHazardousFlag(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCFunctionPlanningFlags(CStringArray &szFlags,CLongUIntArray
          &nFlags) CONST
virtual BOOL CheckTCFunctionPlanningFlag(ULONGLONG nFlag) CONST
virtual INT GetTCFunctionPlanningFlagLength() CONST
```

```
virtual ULONGLONG TranslateTCFunctionPlanningFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionPlanningFlag(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual BOOL CheckTCFunctionAcknowledgementFlags(INT nFlags) CONST
virtual BOOL GetTCFunctionAcknowledgementFlagsRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCFunctionInterlockScopes(CStringArray &szScopes,CLongUIntArray
          &nScopes) CONST
virtual BOOL CheckTCFunctionInterlockScope(ULONGLONG nScope) CONST
virtual INT GetTCFunctionInterlockScopeLength() CONST
virtual ULONGLONG TranslateTCFunctionInterlockScope(LPCTSTR pszScope,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionInterlockScope(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCFunctionInterlockStages(CStringArray &szStages,CLongUIntArray
          &nStages) CONST
virtual BOOL CheckTCFunctionInterlockStage(ULONGLONG nAttributes,ULONGLONG
          nStage) CONST
virtual INT GetTCFunctionInterlockStageLength() CONST
virtual ULONGLONG TranslateTCFunctionInterlockStage(LPCTSTR pszStage,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionInterlockStage(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual BOOL CheckTCFunctionRouteAndDestinationPath(LPCTSTR pszPath) CONST
virtual INT GetTCFunctionRouteAndDestinationPathLength() CONST
virtual BOOL CheckTCFunctionRouterEquipment(LPCTSTR pszEquipment) CONST
virtual INT GetTCFunctionRouterEquipmentLength() CONST
virtual BOOL CheckTCFunctionRouterSubSystem(LPCTSTR pszSubSystem) CONST
virtual INT GetTCFunctionRouterSubSystemLength() CONST
virtual BOOL CheckTCFunctionDestinationEquipment(LPCTSTR pszEquipment) CONST
virtual INT GetTCFunctionDestinationEquipmentLength() CONST
virtual BOOL CheckTCFunctionDestinationSubSystem(LPCTSTR pszSubSystem) CONST
virtual INT GetTCFunctionDestinationSubSystemLength() CONST
virtual BOOL CheckTCFunctionDestinationSubSystemID(INT nSubSystemID) CONST
virtual BOOL GetTCFunctionDestinationSubSystemIDRange(INT &nLow,INT &nHigh)
          CONST
virtual BOOL CheckTCFunctionDestinationSubSchedule(LPCTSTR pszSubSchedule) CONST
virtual INT GetTCFunctionDestinationSubScheduleLength() CONST
virtual BOOL CheckTCFunctionDestinationSubScheduleID(INT nSubScheduleID) CONST
virtual BOOL GetTCFunctionDestinationSubScheduleIDRange(INT &nLow,INT &nHigh)
          CONST
virtual BOOL CheckTCFunctionOperationalInfo(LPCTSTR pszInfo) CONST
virtual INT GetTCFunctionOperationalInfoLength() CONST
virtual BOOL CheckTCFunctionValidityConditions(LPCTSTR pszConditions) CONST
virtual INT GetTCFunctionValidityConditionsLength() CONST
virtual BOOL CheckTCFunctionAssociatedTelecommands(LPCTSTR pszTelecommands)
          CONST
```

```
virtual INT GetTCFunctionAssociatedTelecommandsLength() CONST
virtual BOOL CheckTCFunctionComplementaryTelecommands(LPCTSTR pszTelecommands)
         CONST
virtual INT GetTCFunctionComplementaryTelecommandsLength() CONST
virtual BOOL CheckTCFunctionDefaultParameterSetName(LPCTSTR pszName) CONST
virtual INT GetTCFunctionDefaultParameterSetNameLength() CONST
virtual BOOL CheckTCFunctionMissionPlannerEvent(LPCTSTR pszEvent) CONST
virtual INT GetTCFunctionMissionPlannerEventLength() CONST
virtual BOOL CheckTCFunctionPreExeGroup(LPCTSTR pszGroup) CONST
virtual INT GetTCFunctionPreExeGroupLength() CONST
virtual BOOL CheckTCFunctionExeVerGroup(LPCTSTR pszGroup) CONST
virtual INT GetTCFunctionExeVerGroupLength() CONST
virtual INT EnumTCFunctionStatus(CStringArray &szStatus,CUIntArray &nStatus)
         CONST
virtual BOOL CheckTCFunctionStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCFunctionStatusLength() CONST
virtual INT TranslateTCFunctionStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

Telecommand Function Elements:

```
virtual BOOL CheckTCFunctionElementName(LPCTSTR pszName) CONST
virtual INT GetTCFunctionElementNameLength() CONST
virtual BOOL CheckTCFunctionElementDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCFunctionElementDescriptionLength() CONST
virtual INT EnumTCFunctionElementTypes(CStringArray &szTypes,CLongUIntArray
         &nTypes) CONST
virtual BOOL CheckTCFunctionElementType(LPCTSTR pszName,ULONGLONG nType) CONST
virtual INT GetTCFunctionElementTypeLength() CONST
virtual ULONGLONG TranslateTCFunctionElementType(LPCTSTR pszType,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionElementType(ULONGLONG nAttributes,BOOL
         bDBMS=TRUE) CONST
virtual BOOL CheckTCFunctionElementGroupSize(INT nSize) CONST
virtual BOOL GetTCFunctionElementGroupSizeRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCFunctionElementValueInterpretations(CStringArray
         &szInterpretations,CLongUIntArray &nInterpretations) CONST
virtual BOOL CheckTCFunctionElementValueInterpretation(ULONGLONG
         nAttributes,ULONGLONG nInterpretation) CONST
virtual INT GetTCFunctionElementValueInterpretationLength() CONST
virtual ULONGLONG TranslateTCFunctionElementValueInterpretation(LPCTSTR
         pszInterpretation,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionElementValueInterpretation(ULONGLONG
         nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCFunctionElementValueRadixes(CStringArray
         &szRadixes,CLongUIntArray &nRadixes) CONST
virtual BOOL CheckTCFunctionElementValueRadix(ULONGLONG nAttributes,ULONGLONG
         nRadix) CONST
```

```
virtual INT GetTCFunctionElementValueRadixLength() CONST
virtual ULONGLONG TranslateTCFunctionElementValueRadix(LPCTSTR pszRadix,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionElementValueRadix(ULONGLONG nAttributes,BOOL
         bDBMS=TRUE) CONST
virtual BOOL CheckTCFunctionElementValueBitWidth(INT nWidth) CONST
virtual BOOL GetTCFunctionElementValueBitWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCFunctionElementValueBitOffset(INT nOffset) CONST
virtual BOOL GetTCFunctionElementValueBitOffsetRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCFunctionElementConstValue(ULONGLONG nAttributes,LPCTSTR
         pszConstValue) CONST
virtual INT GetTCFunctionElementConstValueLength() CONST
virtual ULONGLONG TranslateTCFunctionElementConstValue(ULONGLONG nAttributes,INT
         nWidth,LPCTSTR pszConstValue) CONST
virtual CString TranslateTCFunctionElementConstValue(ULONGLONG nAttributes,INT
         nWidth,ULONGLONG nConstValue) CONST
virtual BOOL CheckTCFunctionElementConstValueSource(LPCTSTR pszParameter) CONST
virtual INT GetTCFunctionElementConstValueSourceLength() CONST
virtual BOOL CheckTCFunctionElementConstTimeType(LPCTSTR pszName) CONST
virtual BOOL CheckTCFunctionElementConstTime(ULONGLONG nAttributes,LPCTSTR
         pszConstTime) CONST
virtual INT GetTCFunctionElementConstTimeLength() CONST
virtual CTimeTag TranslateTCFunctionElementConstTime(ULONGLONG
         nAttributes,LPCTSTR pszConstTime) CONST
```

Telecommand Function Blocks:

```
virtual BOOL CheckTCFunctionBlockName(LPCTSTR pszName) CONST
virtual INT GetTCFunctionBlockNameLength() CONST
virtual INT EnumTCFunctionBlockInterpretations(CStringArray
         &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTCFunctionBlockInterpretation(UINT nInterpretation) CONST
virtual INT GetTCFunctionBlockInterpretationLength() CONST
virtual UINT TranslateTCFunctionBlockInterpretation(LPCTSTR
         pszInterpretation,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionBlockInterpretation(UINT nAttributes,BOOL
         bDBMS=TRUE) CONST
virtual BOOL CheckTCFunctionBlockByteOffset(INT nOffset) CONST
virtual BOOL GetTCFunctionBlockByteOffsetRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCFunctionBlockMaxLength(INT nLength) CONST
virtual BOOL GetTCFunctionBlockMaxLengthRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCFunctionBlockStatus(CStringArray &szStatus,CUIntArray
         &nStatus) CONST
virtual BOOL CheckTCFunctionBlockStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCFunctionBlockStatusLength() CONST
virtual INT TranslateTCFunctionBlockStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
         CONST
```

```
virtual CString TranslateTCFunctionBlockStatus(INT nStatus,BOOL bDBMS=TRUE)
        CONST
virtual BOOL CheckTCFunctionBlockElementName(LPCTSTR pszName) CONST
virtual INT GetTCFunctionBlockElementNameLength() CONST
virtual INT EnumTCFunctionBlockElementInterpretations(CStringArray
        &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTCFunctionBlockElementInterpretation(UINT nInterpretation)
        CONST
virtual INT GetTCFunctionBlockElementInterpretationLength() CONST
virtual UINT TranslateTCFunctionBlockElementInterpretation(LPCTSTR
        pszInterpretation,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCFunctionBlockElementInterpretation(UINT
        nAttributes,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.5. Telecommand Sequences

```
virtual BOOL CheckTCSequenceName(LPCTSTR pszName) CONST
virtual INT GetTCSequenceNameLength() CONST
virtual BOOL CheckTCSequenceDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCSequenceDescriptionLength() CONST
virtual BOOL CheckTCSequenceDetails(LPCTSTR pszDetails) CONST
virtual INT GetTCSequenceDetailsLength() CONST
virtual INT EnumTCSequenceTimeTaggedFlags(CStringArray &szFlags,CLongUIntArray
        &nFlags) CONST
virtual BOOL CheckTCSequenceTimeTaggedFlag(ULONGLONG nFlag) CONST
virtual INT GetTCSequenceTimeTaggedFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceTimeTaggedFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceTimeTaggedFlag(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCSequenceHazardousFlags(CStringArray &szFlags,CLongUIntArray
        &nFlags) CONST
virtual BOOL CheckTCSequenceHazardousFlag(ULONGLONG nFlag) CONST
virtual INT GetTCSequenceHazardousFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceHazardousFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceHazardousFlag(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCSequencePlanningFlags(CStringArray &szFlags,CLongUIntArray
        &nFlags) CONST
virtual BOOL CheckTCSequencePlanningFlag(ULONGLONG nFlag) CONST
virtual INT GetTCSequencePlanningFlagLength() CONST
virtual ULONGLONG TranslateTCSequencePlanningFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequencePlanningFlag(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
```

```
virtual INT EnumTCSequenceExecutionFlags(CStringArray &szFlags,CLongUIntArray
        &nFlags) CONST
virtual BOOL CheckTCSequenceExecutionFlag(ULONGLONG nFlag) CONST
virtual INT GetTCSequenceExecutionFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceExecutionFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceExecutionFlag(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckTCSequenceSubSystemID(INT nSubSystemID) CONST
virtual BOOL GetTCSequenceSubSystemIDRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCSequenceSubScheduleID(INT nSubScheduleID) CONST
virtual BOOL GetTCSequenceSubScheduleIDRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCSequenceSubScheduleDestination(LPCTSTR pszSubSchedule) CONST
virtual INT GetTCSequenceSubScheduleDestinationLength() CONST
virtual BOOL CheckTCSequenceDefaultParameterSetName(LPCTSTR pszName) CONST
virtual INT GetTCSequenceDefaultParameterSetNameLength() CONST
virtual BOOL CheckTCSequenceMissionPlannerEvent(LPCTSTR pszEvent) CONST
virtual INT GetTCSequenceMissionPlannerEventLength() CONST
virtual BOOL CheckTCSequenceUserName(LPCTSTR pszUserName) CONST
virtual INT GetTCSequenceUserNameLength() CONST
virtual BOOL CheckTCSequenceDate(LPCTSTR pszDate,LPCTSTR pszFormat) CONST
virtual INT GetTCSequenceDateLength() CONST
virtual CTimeKey TranslateTCSequenceDate(LPCTSTR pszDate,LPCTSTR pszFormat)
        CONST
virtual CString TranslateTCSequenceDate(CONST CTimeKey &tDate,LPCTSTR pszFormat)
        CONST
virtual BOOL CheckTCSequenceDocumentName(LPCTSTR pszDocumentName) CONST
virtual INT GetTCSequenceDocumentNameLength() CONST
virtual BOOL CheckTCSequenceDocumentIssue(LPCTSTR pszDocumentIssue) CONST
virtual INT GetTCSequenceDocumentIssueLength() CONST
virtual BOOL CheckTCSequenceDocumentDate(LPCTSTR pszDate,LPCTSTR pszFormat)
        CONST
virtual INT GetTCSequenceDocumentDateLength() CONST
virtual CTimeKey TranslateTCSequenceDocumentDate(LPCTSTR pszDate,LPCTSTR
        pszFormat) CONST
virtual CString TranslateTCSequenceDocumentDate(CONST CTimeKey &tDate,LPCTSTR
        pszFormat) CONST
virtual BOOL CheckTCSequenceEntryCount(INT nCount) CONST
virtual BOOL GetTCSequenceEntryCountRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCSequenceParameterCount(INT nCount) CONST
virtual BOOL GetTCSequenceParameterCountRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCSequenceStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
virtual BOOL CheckTCSequenceStatus(LPCTSTR pszStatus) CONST
```

```
virtual INT GetTCSequenceStatusLength() CONST
virtual INT TranslateTCSequenceStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

Telecommand Sequence Parameters:

```
virtual BOOL CheckTCSequenceParameterTag(LPCTSTR pszTag) CONST
virtual INT GetTCSequenceParameterTagLength() CONST
virtual BOOL CheckTCSequenceParameterDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCSequenceParameterDescriptionLength() CONST
virtual BOOL CheckTCSequenceParameterUnit(LPCTSTR pszUnit) CONST
virtual INT GetTCSequenceParameterUnitLength() CONST
virtual BOOL CheckTCSequenceParameterPosition(INT nPosition) CONST
virtual BOOL GetTCSequenceParameterPositionRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCSequenceParameterTypes(CStringArray &szTypes,CLongUIntArray
          &nTypes) CONST
virtual BOOL CheckTCSequenceParameterType(ULONGLONG nType) CONST
virtual INT GetTCSequenceParameterTypeLength() CONST
virtual ULONGLONG TranslateTCSequenceParameterType(LPCTSTR pszType,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceParameterType(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCSequenceParameterCategories(CStringArray
          &szCategories,CLongUIntArray &nCategories) CONST
virtual BOOL CheckTCSequenceParameterCategory(ULONGLONG nCategory) CONST
virtual INT GetTCSequenceParameterCategoryLength() CONST
virtual ULONGLONG TranslateTCSequenceParameterCategory(LPCTSTR pszCategory,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceParameterCategory(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCSequenceParameterTypeCodes(CStringArray &szPTCs,CUIntArray
          &nPTCs) CONST
virtual BOOL CheckTCSequenceParameterTypeCode(INT nPTC) CONST
virtual BOOL GetTCSequenceParameterTypeCodeRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTCSequenceParameterTypeCode(LPCTSTR pszPTC) CONST
virtual CString TranslateTCSequenceParameterTypeCode(INT nPTC,INT nPFC) CONST
virtual INT EnumTCSequenceParameterFormatCodes(LPCTSTR pszPTC,CStringArray
          &szPFCs,CUIntArray &nPFCs) CONST
virtual BOOL CheckTCSequenceParameterFormatCode(INT nPTC,INT nPFC) CONST
virtual BOOL GetTCSequenceParameterFormatCodeRange(INT nPTC,INT &nLow,INT
          &nHigh) CONST
virtual INT TranslateTCSequenceParameterFormatCode(LPCTSTR pszPFC) CONST
virtual CString TranslateTCSequenceParameterFormatCode(INT nPTC,INT nPFC) CONST
virtual INT EnumTCSequenceParameterDisplayFormats(CStringArray
          &szFormats,CLongUIntArray &nFormats) CONST
virtual INT EnumTCSequenceParameterDisplayFormats(ULONGLONG
          nAttributes,CStringArray &szFormats,CLongUIntArray &nFormats) CONST
```

```
virtual BOOL CheckTCSequenceParameterDisplayFormat(ULONGLONG
        nAttributes,ULONGLONG nFormat) CONST
virtual INT GetTCSequenceParameterDisplayFormatLength() CONST
virtual ULONGLONG TranslateTCSequenceParameterDisplayFormat(LPCTSTR
        pszFormat,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceParameterDisplayFormat(ULONGLONG
        nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCSequenceParameterCodings(CStringArray
        &szCodings,CLongUIntArray &nCodings) CONST
virtual BOOL CheckTCSequenceParameterCoding(ULONGLONG nAttributes,ULONGLONG
        nCoding) CONST
virtual INT GetTCSequenceParameterCodingLength() CONST
virtual ULONGLONG TranslateTCSequenceParameterCoding(LPCTSTR pszCoding,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceParameterCoding(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCSequenceParameterValueTypes(CStringArray
        &szTypes,CLongUIntArray &nTypes) CONST
virtual BOOL CheckTCSequenceParameterValueType(ULONGLONG nAttributes,ULONGLONG
        nType) CONST
virtual INT GetTCSequenceParameterValueTypeLength() CONST
virtual ULONGLONG TranslateTCSequenceParameterValueType(LPCTSTR pszType,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceParameterValueType(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCSequenceParameterValueRadixes(CStringArray
        &szRadixes,CLongUIntArray &nRadixes) CONST
virtual BOOL CheckTCSequenceParameterValueRadix(ULONGLONG nAttributes,ULONGLONG
        nRadix) CONST
virtual INT GetTCSequenceParameterValueRadixLength() CONST
virtual ULONGLONG TranslateTCSequenceParameterValueRadix(LPCTSTR pszRadix,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceParameterValueRadix(ULONGLONG
        nAttributes,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTCSequenceParameterBitWidth(INT nWidth) CONST
virtual BOOL GetTCSequenceParameterBitWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCSequenceParameterCalTable(ULONGLONG nAttributes,LPCTSTR
        pszTable) CONST
virtual INT GetTCSequenceParameterCalTableLength() CONST
virtual BOOL CheckTCSequenceParameterOolTable(ULONGLONG nAttributes,LPCTSTR
        pszTable) CONST
virtual INT GetTCSequenceParameterOolTableLength() CONST
virtual BOOL CheckTCSequenceParameterConstValue(ULONGLONG nAttributes,LPCTSTR
        pszConstValue) CONST
virtual INT GetTCSequenceParameterConstValueLength() CONST
```
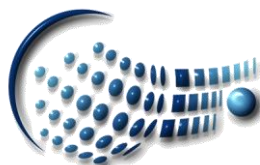
```
virtual ULONGLONG TranslateTCSequenceParameterConstValue(ULONGLONG
        nAttributes,INT nWidth,LPCTSTR pszConstValue) CONST
virtual CString TranslateTCSequenceParameterConstValue(ULONGLONG nAttributes,INT
        nWidth,ULONGLONG nConstValue) CONST
virtual BOOL CheckTCSequenceParameterConstTimeType(LPCTSTR pszTag) CONST
virtual BOOL CheckTCSequenceParameterConstTime(ULONGLONG nAttributes,LPCTSTR
        pszConstTime) CONST
virtual INT GetTCSequenceParameterConstTimeLength() CONST
virtual CTimeTag TranslateTCSequenceParameterConstTime(ULONGLONG
        nAttributes,LPCTSTR pszConstTime) CONST
```

Telecommand Sequence Entries:

```
virtual BOOL CheckTCSequenceEntryName(LPCTSTR pszName) CONST
virtual INT GetTCSequenceEntryNameLength() CONST
virtual BOOL CheckTCSequenceEntryDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCSequenceEntryDescriptionLength() CONST
virtual INT EnumTCSequenceEntryTypes(CStringArray &szTypes,CLongUIntArray
        &nTypes) CONST
virtual BOOL CheckTCSequenceEntryType(ULONGLONG nType) CONST
virtual INT GetTCSequenceEntryTypeLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryType(LPCTSTR pszType,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCSequenceEntryType(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCSequenceEntrySubTypes(CStringArray &szSubTypes,CLongUIntArray
        &nSubTypes) CONST
virtual BOOL CheckTCSequenceEntrySubType(ULONGLONG nSubType) CONST
virtual INT GetTCSequenceEntrySubTypeLength() CONST
virtual ULONGLONG TranslateTCSequenceEntrySubType(LPCTSTR pszSubType,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntrySubType(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckTCSequenceEntryIndex(INT nIndex) CONST
virtual BOOL GetTCSequenceEntryIndexRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCSequenceEntry1Step(LPCTSTR psz1Step) CONST
virtual INT GetTCSequenceEntry1StepLength() CONST
virtual INT EnumTCSequenceEntryDispatchModes(CStringArray
        &szModes,CLongUIntArray &nModes) CONST
virtual BOOL CheckTCSequenceEntryDispatchMode(ULONGLONG nAttributes,ULONGLONG
        nMode) CONST
virtual INT GetTCSequenceEntryDispatchModeLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryDispatchMode(LPCTSTR pszMode,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryDispatchMode(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCSequenceEntryReleaseTimeTypes(CStringArray
        &szTypes,CLongUIntArray &nTypes) CONST
```

```
virtual BOOL CheckTCSequenceEntryReleaseTimeType(ULONGLONG nAttributes,ULONGLONG
            nType) CONST
virtual INT GetTCSequenceEntryReleaseTimeTypeLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryReleaseTimeType(LPCTSTR pszType,BOOL
            bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryReleaseTimeType(ULONGLONG
            nAttributes,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTCSequenceEntryReleaseDeltaTime(ULONGLONG nAttributes,LPCTSTR
            pszDelta,LPCTSTR pszFormat) CONST
virtual INT GetTCSequenceEntryReleaseDeltaTimeLength() CONST
virtual DWORD TranslateTCSequenceEntryReleaseDeltaTime(LPCTSTR pszDelta,LPCTSTR
            pszFormat) CONST
virtual CString TranslateTCSequenceEntryReleaseDeltaTime(DWORD dwDelta,LPCTSTR
            pszFormat) CONST
virtual INT EnumTCSequenceEntryExecutionTimeTypes(CStringArray
            &szTypes,CLongUIntArray &nTypes) CONST
virtual BOOL CheckTCSequenceEntryExecutionTimeType(ULONGLONG
            nAttributes,ULONGLONG nType) CONST
virtual INT GetTCSequenceEntryExecutionTimeTypeLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryExecutionTimeType(LPCTSTR pszType,BOOL
            bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryExecutionTimeType(ULONGLONG
            nAttributes,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTCSequenceEntryExecutionTime(LPCTSTR pszTime,LPCTSTR
            pszFormat) CONST
virtual INT GetTCSequenceEntryExecutionTimeLength() CONST
virtual CTimeTag TranslateTCSequenceEntryExecutionTime(LPCTSTR pszTime,LPCTSTR
            pszFormat) CONST
virtual CString TranslateTCSequenceEntryExecutionTime(CONST CTimeTag
            &tTime,LPCTSTR pszFormat) CONST
virtual BOOL CheckTCSequenceEntryExecutionDeltaTime(LPCTSTR pszDelta,LPCTSTR
            pszFormat) CONST
virtual INT GetTCSequenceEntryExecutionDeltaTimeLength() CONST
virtual DWORD TranslateTCSequenceEntryExecutionDeltaTime(LPCTSTR pszDelta,BOOL
            &bEarlier,LPCTSTR pszFormat) CONST
virtual CString TranslateTCSequenceEntryExecutionDeltaTime(DWORD dwDelta,BOOL
            bEarlier,LPCTSTR pszFormat) CONST
virtual INT EnumTCSequenceEntryGroupFlags(CStringArray &szFlags,CLongUIntArray
            &nFlags) CONST
virtual BOOL CheckTCSequenceEntryGroupFlag(ULONGLONG nAttributes,ULONGLONG
            nFlag) CONST
virtual INT GetTCSequenceEntryGroupFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryGroupFlag(LPCTSTR pszFlag,BOOL
            bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryGroupFlag(ULONGLONG nAttributes,BOOL
            bDBMS=TRUE) CONST
```

```
virtual INT EnumTCSequenceEntryBlockFlags(CStringArray &szFlags,CLongUIntArray
          &nFlags) CONST
virtual BOOL CheckTCSequenceEntryBlockFlag(ULONGLONG nAttributes,ULONGLONG
          nFlag) CONST
virtual INT GetTCSequenceEntryBlockFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryBlockFlag(LPCTSTR pszFlag,LPCTSTR
          pszMember,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryBlockFlag(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual BOOL CheckTCSequenceEntryBlockMembership(LPCTSTR pszMember) CONST
virtual INT GetTCSequenceEntryBlockMembershipLength() CONST
virtual INT TranslateTCSequenceEntryBlockMembership(LPCTSTR pszMember) CONST
virtual CString TranslateTCSequenceEntryBlockMembership(INT nMember) CONST
virtual INT EnumTCSequenceEntryInterlockScopes(CStringArray
          &szScopes,CLongUIntArray &nScopes) CONST
virtual BOOL CheckTCSequenceEntryInterlockScope(ULONGLONG nAttributes,ULONGLONG
          nScope) CONST
virtual INT GetTCSequenceEntryInterlockScopeLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryInterlockScope(LPCTSTR pszScope,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryInterlockScope(ULONGLONG
          nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCSequenceEntryInterlockStages(CStringArray
          &szStages,CLongUIntArray &nStages) CONST
virtual BOOL CheckTCSequenceEntryInterlockStage(ULONGLONG nAttributes,ULONGLONG
          nStage) CONST
virtual INT GetTCSequenceEntryInterlockStageLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryInterlockStage(LPCTSTR pszStage,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryInterlockStage(ULONGLONG
          nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCSequenceEntryPTVCheckFlags(CStringArray
          &szFlags,CLongUIntArray &nFlags) CONST
virtual BOOL CheckTCSequenceEntryPTVCheckFlag(ULONGLONG nAttributes,ULONGLONG
          nFlag) CONST
virtual INT GetTCSequenceEntryPTVCheckFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryPTVCheckFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryPTVCheckFlag(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCSequenceEntryPEVCheckFlags(CStringArray
          &szFlags,CLongUIntArray &nFlags) CONST
virtual BOOL CheckTCSequenceEntryPEVCheckFlag(ULONGLONG nAttributes,ULONGLONG
          nFlag) CONST
```

```
virtual INT GetTCSequenceEntryPEVCheckFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryPEVCheckFlag(LPCTSTR pszFlag,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryPEVCheckFlag(ULONGLONG nAttributes,BOOL
         bDBMS=TRUE) CONST
virtual INT EnumTCSequenceEntryCEVCheckFlags(CStringArray
         &szFlags,CLongUIntArray &nFlags) CONST
virtual BOOL CheckTCSequenceEntryCEVCheckFlag(ULONGLONG nAttributes,ULONGLONG
         nFlag) CONST
virtual INT GetTCSequenceEntryCEVCheckFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryCEVCheckFlag(LPCTSTR pszFlag,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryCEVCheckFlag(ULONGLONG nAttributes,BOOL
         bDBMS=TRUE) CONST
virtual BOOL CheckTCSequenceEntryParameterCount(INT nCount) CONST
virtual BOOL GetTCSequenceEntryParameterCountRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCSequenceEntryParameterTag(LPCTSTR pszTag) CONST
virtual INT GetTCSequenceEntryParameterTagLength() CONST
virtual BOOL CheckTCSequenceEntryParameterComment(LPCTSTR pszTag) CONST
virtual INT GetTCSequenceEntryParameterCommentLength() CONST
virtual BOOL CheckTCSequenceEntryParameterPosition(INT nPosition) CONST
virtual BOOL GetTCSequenceEntryParameterPositionRange(INT &nLow,INT &nHigh)
         CONST
virtual INT EnumTCSequenceEntryParameterModificationFlags(CStringArray
         &szFlags,CLongUIntArray &nFlags) CONST
virtual BOOL CheckTCSequenceEntryParameterModificationFlag(ULONGLONG nFlag)
         CONST
virtual INT GetTCSequenceEntryParameterModificationFlagLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryParameterModificationFlag(LPCTSTR
         pszFlag,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryParameterModificationFlag(ULONGLONG
         nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCSequenceEntryParameterValueTypes(CStringArray
         &szTypes,CLongUIntArray &nTypes) CONST
virtual BOOL CheckTCSequenceEntryParameterValueType(ULONGLONG nType) CONST
virtual INT GetTCSequenceEntryParameterValueTypeLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryParameterValueType(LPCTSTR
         pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryParameterValueType(ULONGLONG
         nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCSequenceEntryParameterValueRadixes(CStringArray
         &szRadixes,CLongUIntArray &nRadixes) CONST
virtual BOOL CheckTCSequenceEntryParameterValueRadix(ULONGLONG
         nAttributes,ULONGLONG nRadix) CONST
virtual INT GetTCSequenceEntryParameterValueRadixLength() CONST
```

```
virtual ULONGLONG TranslateTCSequenceEntryParameterValueRadix(LPCTSTR
        pszRadix,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCSequenceEntryParameterValueRadix(ULONGLONG
        nAttributes,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTCSequenceEntryParameterBitOffset(INT nOffset) CONST
virtual BOOL GetTCSequenceEntryParameterBitOffsetRange(INT &nLow,INT &nHigh)
        CONST
virtual BOOL CheckTCSequenceEntryParameterConstValue(ULONGLONG
        nAttributes,LPCTSTR pszConstValue) CONST
virtual INT GetTCSequenceEntryParameterConstValueLength() CONST
virtual ULONGLONG TranslateTCSequenceEntryParameterConstValue(ULONGLONG
        nAttributes,INT nWidth,LPCTSTR pszConstValue) CONST
virtual CString TranslateTCSequenceEntryParameterConstValue(ULONGLONG
        nAttributes,INT nWidth,ULONGLONG nConstValue) CONST
virtual BOOL CheckTCSequenceEntryParameterConstTimeType(LPCTSTR pszTag) CONST
virtual BOOL CheckTCSequenceEntryParameterConstTime(ULONGLONG
        nAttributes,LPCTSTR pszConstTime) CONST
virtual INT GetTCSequenceEntryParameterConstTimeLength() CONST
virtual CTimeTag TranslateTCSequenceEntryParameterConstTime(ULONGLONG
        nAttributes,LPCTSTR pszConstTime) CONST
virtual BOOL CheckTCSequenceEntryParameterValueSetName(LPCTSTR pszName) CONST
virtual INT GetTCSequenceEntryParameterValueSetNameLength() CONST
virtual BOOL CheckTCSequenceEntryParameterMemoryAreaName(LPCTSTR pszName) CONST
virtual INT GetTCSequenceEntryParameterMemoryAreaNameLength() CONST
virtual BOOL CheckTCSequenceEntryParameterMemoryAreaVersion(LPCTSTR pszVersion)
        CONST
virtual INT GetTCSequenceEntryParameterMemoryAreaVersionLength() CONST
```

3.1.2.5.6. Telecommand Procedures

```
virtual BOOL CheckTCProcedureName(LPCTSTR pszName) CONST
virtual INT GetTCProcedureNameLength() CONST
virtual BOOL CheckTCProcedureDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCProcedureDescriptionLength() CONST
virtual BOOL CheckTCProcedureCreationDate(CONST CTimeKey &tDate) CONST
virtual BOOL CheckTCProcedureModificationDate(CONST CTimeKey
        &tCreationDate,CONST CTimeKey &tModificationDate) CONST
virtual BOOL CheckTCProcedureExecutionMode(UINT nMode) CONST
virtual BOOL CheckTCProcedureArgumentValue(ULONGLONG nAttributes,LPCTSTR
        pszValue) CONST
virtual ULONGLONG TranslateTCProcedureArgumentValue(ULONGLONG nAttributes,INT
        nWidth,LPCTSTR pszValue) CONST
virtual CString TranslateTCProcedureArgumentValue(ULONGLONG nAttributes,INT
        nWidth,ULONGLONG nValue) CONST
virtual CTimeTag TranslateTCProcedureArgumentTime(ULONGLONG nAttributes,LPCTSTR
        pszTime) CONST
```

```
virtual BOOL CheckTCProcedureVariableValue(ULONGLONG nAttributes,LPCTSTR
        pszValue) CONST
virtual ULONGLONG TranslateTCProcedureVariableValue(ULONGLONG nAttributes,INT
        nWidth,LPCTSTR pszValue) CONST
virtual CString TranslateTCProcedureVariableValue(ULONGLONG nAttributes,INT
        nWidth,ULONGLONG nValue) CONST
virtual CTimeTag TranslateTCProcedureVariableTime(ULONGLONG nAttributes,LPCTSTR
        pszTime) CONST
```

3.1.2.5.7. On-Board Processors

```
virtual BOOL CheckOBProcessorName(LPCTSTR pszName) CONST
virtual INT GetOBProcessorNameLength() CONST
virtual BOOL CheckOBProcessorDescription(LPCTSTR pszDescription) CONST
virtual INT GetOBProcessorDescriptionLength() CONST
virtual INT EnumOBProcessorAddressBases(CStringArray &szBases,CUIntArray
        &nBases) CONST
virtual BOOL CheckOBProcessorAddressBase(UINT nBase) CONST
virtual INT GetOBProcessorAddressBaseLength() CONST
virtual UINT TranslateOBProcessorAddressBase(LPCTSTR pszBase,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateOBProcessorAddressBase(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckOBProcessorPatchCommand(LPCTSTR pszCommand) CONST
virtual INT GetOBProcessorPatchCommandLength() CONST
virtual BOOL CheckOBProcessorDumpCommand(LPCTSTR pszCommand) CONST
virtual INT GetOBProcessorDumpCommandLength() CONST
virtual BOOL CheckOBProcessorDumpPacket(LPCTSTR pszPacket) CONST
virtual INT GetOBProcessorDumpPacketLength() CONST
virtual BOOL CheckOBProcessorDumpInterval(CONST CTimeSpan &tInterval) CONST
virtual BOOL GetOBProcessorDumpIntervalRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckOBProcessorDumpCount(INT nCount) CONST
virtual BOOL GetOBProcessorDumpCountRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckOBProcessorMasterImage(LPCTSTR pszImage) CONST
virtual INT GetOBProcessorMasterImageLength() CONST
virtual UINT GetOBProcessorAddressWidth() CONST
virtual BOOL CheckOBProcessorAddressRange(UINT nStartPage,UINT
        nStartAddress,UINT nEndPage,UINT nEndAddress) CONST
virtual BOOL GetOBProcessorAddressRange(UINT &nPageLow,UINT &nPageHigh,UINT
        &nAddressLow,UINT &nAddressHigh) CONST
virtual BOOL TranslateOBProcessorAddressRange(LPCTSTR pszAddressRange,UINT
        &nStartPage,UINT &nStartAddress,UINT &nEndPage,UINT &nEndAddress)
        CONST
virtual CString TranslateOBProcessorAddressRange(UINT nStartPage,UINT
        nStartAddress,UINT nEndPage,UINT nEndAddress) CONST
virtual INT EnumOBProcessorStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
```

```
virtual BOOL CheckOBProcessorStatus(LPCTSTR pszStatus) CONST
virtual INT GetOBProcessorStatusLength() CONST
virtual INT TranslateOBProcessorStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateOBProcessorStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

On-Board Memory Areas:

```
virtual BOOL CheckOBProcessorMemoryAreaName(LPCTSTR pszName) CONST
virtual INT GetOBProcessorMemoryAreaNameLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaDescription(LPCTSTR pszDescription) CONST
virtual INT GetOBProcessorMemoryAreaDescriptionLength() CONST
virtual INT EnumOBProcessorMemoryAreaTypes(CStringArray &szTypes,CUIntArray
        &nTypes) CONST
virtual BOOL CheckOBProcessorMemoryAreaType(UINT nType) CONST
virtual INT GetOBProcessorMemoryAreaTypeLength() CONST
virtual UINT TranslateOBProcessorMemoryAreaType(LPCTSTR pszType,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateOBProcessorMemoryAreaType(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckOBProcessorMemoryAreaPatchCommand(LPCTSTR pszCommand) CONST
virtual INT GetOBProcessorMemoryAreaPatchCommandLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaDumpCommand(LPCTSTR pszCommand) CONST
virtual INT GetOBProcessorMemoryAreaDumpCommandLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaDumpPacket(LPCTSTR pszPacket) CONST
virtual INT GetOBProcessorMemoryAreaDumpPacketLength() CONST
virtual INT EnumOBProcessorMemoryAreaPatchFlags(CStringArray &szFlags,CUIntArray
        &nFlags) CONST
virtual BOOL CheckOBProcessorMemoryAreaPatchFlag(UINT nFlag) CONST
virtual INT GetOBProcessorMemoryAreaPatchFlagLength() CONST
virtual UINT TranslateOBProcessorMemoryAreaPatchFlag(LPCTSTR szFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateOBProcessorMemoryAreaPatchFlag(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumOBProcessorMemoryAreaDumpFlags(CStringArray &szFlags,CUIntArray
        &nFlags) CONST
virtual BOOL CheckOBProcessorMemoryAreaDumpFlag(UINT nFlag) CONST
virtual INT GetOBProcessorMemoryAreaDumpFlagLength() CONST
virtual UINT TranslateOBProcessorMemoryAreaDumpFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateOBProcessorMemoryAreaDumpFlag(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumOBProcessorMemoryAreaUpdateFlags(CStringArray
        &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckOBProcessorMemoryAreaUpdateFlag(UINT nFlag) CONST
virtual INT GetOBProcessorMemoryAreaUpdateFlagLength() CONST
```

```
virtual UINT TranslateOBProcessorMemoryAreaUpdateFlag(LPCTSTR pszFlag,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateOBProcessorMemoryAreaUpdateFlag(UINT nAttributes,BOOL
         bDBMS=TRUE) CONST
virtual INT EnumOBProcessorMemoryAreaInhibitFlags(CStringArray
         &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckOBProcessorMemoryAreaInhibitFlag(UINT nFlag) CONST
virtual INT GetOBProcessorMemoryAreaInhibitFlagLength() CONST
virtual UINT TranslateOBProcessorMemoryAreaInhibitFlag(LPCTSTR pszFlag,BOOL
         bDBMS=TRUE) CONST
virtual CString TranslateOBProcessorMemoryAreaInhibitFlag(UINT nAttributes,BOOL
         bDBMS=TRUE) CONST
virtual BOOL CheckOBProcessorMemoryAreaBlockName(LPCTSTR pszName) CONST
virtual INT GetOBProcessorMemoryAreaBlockNameLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaBlockDescription(LPCTSTR pszDescription)
         CONST
virtual INT GetOBProcessorMemoryAreaBlockDescriptionLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaBlockTMParameter(LPCTSTR pszTag) CONST
virtual INT GetOBProcessorMemoryAreaBlockTMParameterLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaBlockTCParameter(LPCTSTR pszTag) CONST
virtual INT GetOBProcessorMemoryAreaBlockTCParameterLength() CONST
virtual UINT GetOBProcessorMemoryAreaBlockAddressWidth() CONST
virtual BOOL CheckOBProcessorMemoryAreaBlockAddress(UINT nPage,UINT nAddress)
         CONST
virtual BOOL GetOBProcessorMemoryAreaBlockAddressRange(UINT &nPageLow,UINT
         &nPageHigh,UINT &nAddressLow,UINT &nAddressHigh) CONST
virtual BOOL TranslateOBProcessorMemoryAreaBlockAddress(LPCTSTR pszAddress,UINT
         &nPage,UINT &nAddress) CONST
virtual CString TranslateOBProcessorMemoryAreaBlockAddress(UINT nPage,UINT
         nAddress) CONST
virtual BOOL CheckOBProcessorMemoryAreaBlockBitOffset(INT nOffset) CONST
virtual BOOL GetOBProcessorMemoryAreaBlockBitOffsetRange(INT &nLow,INT &nHigh)
         CONST
virtual BOOL CheckOBProcessorMemoryAreaBlockBitWidth(INT nWidth) CONST
virtual BOOL GetOBProcessorMemoryAreaBlockBitWidthRange(INT &nLow,INT &nHigh)
         CONST
virtual BOOL CheckOBProcessorMemoryAreaCheckSumName(LPCTSTR pszName) CONST
virtual INT GetOBProcessorMemoryAreaCheckSumNameLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaCheckSumSeedValue(LPCTSTR pszSeedValue)
         CONST
virtual INT GetOBProcessorMemoryAreaCheckSumSeedValueLength() CONST
virtual BOOL GetOBProcessorMemoryAreaCheckSumSeedValueRange(UINT &nLow,UINT
         &nHigh) CONST
virtual UINT TranslateOBProcessorMemoryAreaCheckSumSeedValue(LPCTSTR
         pszSeedValue) CONST
```

```
virtual CString TranslateOBProcessorMemoryAreaCheckSumSeedValue(UINT nSeedValue)
          CONST
virtual BOOL CheckOBProcessorMemoryAreaCheckSumLength(INT nLength) CONST
virtual BOOL GetOBProcessorMemoryAreaCheckSumLengthRange(INT &nLow,INT &nHigh)
          CONST
virtual BOOL CheckOBProcessorMemoryAreaCheckSumDestination(UINT nDestination)
          CONST
virtual BOOL GetOBProcessorMemoryAreaCheckSumDestinationRange(UINT &nLow,UINT
          &nHigh) CONST
virtual UINT GetOBProcessorMemoryAreaCheckSumAddressWidth() CONST
virtual BOOL CheckOBProcessorMemoryAreaCheckSumAddressRange(UINT nPage,UINT
          nAddressLow,UINT nAddressHigh) CONST
virtual BOOL GetOBProcessorMemoryAreaCheckSumAddressRange(UINT &nPageLow,UINT
          &nPageHigh,UINT &nAddressLow,UINT &nAddressHigh) CONST
virtual BOOL CheckOBProcessorMemoryAreaCatalogueName(LPCTSTR pszName) CONST
virtual INT GetOBProcessorMemoryAreaCatalogueNameLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaCatalogueEntryID(LPCTSTR pszID) CONST
virtual INT GetOBProcessorMemoryAreaCatalogueEntryIDLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaCatalogueEntryDescription(LPCTSTR
          pszDescription) CONST
virtual INT GetOBProcessorMemoryAreaCatalogueEntryDescriptionLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaCatalogueEntryVersion(LPCTSTR pszVersion)
          CONST
virtual INT GetOBProcessorMemoryAreaCatalogueEntryVersionLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaCatalogueEntryDate(LPCTSTR pszDate) CONST
virtual INT GetOBProcessorMemoryAreaCatalogueEntryDateLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaCatalogueEntryTime(LPCTSTR pszTime) CONST
virtual INT GetOBProcessorMemoryAreaCatalogueEntryTimeLength() CONST
virtual BOOL CheckOBProcessorMemoryAreaCatalogueEntryLocation(LPCTSTR
          pszLocation) CONST
virtual INT GetOBProcessorMemoryAreaCatalogueEntryLocationLength() CONST
virtual INT EnumOBProcessorMemoryAreaCatalogueEntryFlags(CStringArray
          &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckOBProcessorMemoryAreaCatalogueEntryFlag(UINT nFlag) CONST
virtual INT GetOBProcessorMemoryAreaCatalogueEntryFlagLength() CONST
virtual UINT TranslateOBProcessorMemoryAreaCatalogueEntryFlag(LPCTSTR
          pszFlag,BOOL bDBMS=TRUE) CONST
virtual CString TranslateOBProcessorMemoryAreaCatalogueEntryFlag(UINT
          nAttributes,BOOL bDBMS=TRUE) CONST
```

On-Board Memory Images:

```
virtual BOOL CheckOBProcessorMemoryImageName(LPCTSTR pszName) CONST
virtual INT GetOBProcessorMemoryImageNameLength() CONST
virtual BOOL CheckOBProcessorMemoryImageBlockVersion(LPCTSTR pszVersion) CONST
virtual INT GetOBProcessorMemoryImageBlockVersionLength() CONST
virtual UINT GetOBProcessorMemoryImageBlockAddressWidth() CONST
```

```
virtual BOOL CheckOBProcessorMemoryImageBlockAddress(UINT nPage,UINT nAddress)
        CONST
virtual BOOL GetOBProcessorMemoryImageBlockAddressRange(UINT &nPageLow,UINT
        &nPageHigh,UINT &nAddressLow,UINT &nAddressHigh) CONST
virtual BOOL TranslateOBProcessorMemoryImageBlockAddress(LPCTSTR pszAddress,UINT
        &nPage,UINT &nAddress) CONST
virtual CString TranslateOBProcessorMemoryImageBlockAddress(UINT nPage,UINT
        nAddress) CONST
virtual BOOL CheckOBProcessorMemoryImageBlockData(LPCTSTR pszData,BOOL
        bNumbersOnly=FALSE) CONST
virtual BOOL CheckOBProcessorMemoryImageBlockData(CONST CByteArray &nData,CONST
        CByteArray &nMask,BOOL bAllowTemplates=FALSE) CONST
virtual INT GetOBProcessorMemoryImageBlockDataLength() CONST
virtual INT TranslateOBProcessorMemoryImageBlockData(LPCTSTR pszData,CByteArray
        &nData,CByteArray &nMask,BOOL bDBMS=TRUE) CONST
virtual CString TranslateOBProcessorMemoryImageBlockData(CONST CByteArray
        &nData,CONST CByteArray &nMask,BOOL bDBMS=TRUE) CONST
virtual CString TranslateOBProcessorMemoryImageBlockData(LPCTSTR pszData,BOOL
        bDBMS=TRUE) CONST
```

3.1.2.5.8. Telemetry Parameters

```
virtual BOOL CheckTMParameterTag(LPCTSTR pszTag) CONST
virtual INT GetTMParameterTagLength() CONST
virtual BOOL CheckTMParameterDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMParameterDescriptionLength() CONST
virtual BOOL CheckTMParameterDetails(LPCTSTR pszDetails) CONST
virtual INT GetTMParameterDetailsLength() CONST
virtual BOOL CheckTMParameterUnit(LPCTSTR pszUnit) CONST
virtual INT GetTMParameterUnitLength() CONST
virtual INT EnumTMParameterTypes(CStringArray &szTypes,CLongUIntArray &nTypes)
        CONST
virtual BOOL CheckTMParameterType(ULONGLONG nType) CONST
virtual INT GetTMParameterTypeLength() CONST
virtual ULONGLONG TranslateTMParameterType(LPCTSTR pszType,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTMParameterType(ULONGLONG nAttributes,BOOL bDBMS=TRUE)
        CONST
virtual INT EnumTMParameterCategories(CStringArray &szCategories,CLongUIntArray
        &nCategories) CONST
virtual BOOL CheckTMParameterCategory(ULONGLONG nCategory) CONST
virtual INT GetTMParameterCategoryLength() CONST
virtual ULONGLONG TranslateTMParameterCategory(LPCTSTR pszCategory,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMParameterCategory(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
```

```
virtual INT EnumTMParameterNatures(CStringArray &szNatures,CLongUIntArray
          &nNatures) CONST
virtual BOOL CheckTMParameterNature(ULONGLONG nNature) CONST
virtual INT GetTMParameterNatureLength() CONST
virtual ULONGLONG TranslateTMParameterNature(LPCTSTR pszNature,BOOL bDBMS=TRUE)
          CONST
virtual CString TranslateTMParameterNature(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTMParameterTypeCodes(CStringArray &szPTCs,CUIntArray &nPTCs)
          CONST
virtual BOOL CheckTMParameterTypeCode(INT nPTC) CONST
virtual BOOL GetTMParameterTypeCodeRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMParameterTypeCode(LPCTSTR pszPTC) CONST
virtual CString TranslateTMParameterTypeCode(INT nPTC,INT nPFC) CONST
virtual INT EnumTMParameterFormatCodes(LPCTSTR pszPTC,CStringArray
          &szPFCs,CUIntArray &nPFCs) CONST
virtual BOOL CheckTMParameterFormatCode(INT nPTC,INT nPFC) CONST
virtual BOOL GetTMParameterFormatCodeRange(INT nPTC,INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMParameterFormatCode(LPCTSTR pszPFC) CONST
virtual CString TranslateTMParameterFormatCode(INT nPTC,INT nPFC) CONST
virtual BOOL CheckTMParameterBitWidth(ULONGLONG nAttributes,INT nWidth) CONST
virtual BOOL GetTMParameterBitWidthRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTMParameterOBSTypes(CStringArray &szOBSTypes,CLongUIntArray
          &nOBSTypes) CONST
virtual BOOL CheckTMParameterOBSType(ULONGLONG nAttributes,ULONGLONG nOBSType)
          CONST
virtual INT GetTMParameterOBSTypeLength() CONST
virtual ULONGLONG TranslateTMParameterOBSType(LPCTSTR pszOBSType,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMParameterOBSType(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTMParameterCodings(CStringArray &szCodings,CLongUIntArray
          &nCodings) CONST
virtual BOOL CheckTMParameterCoding(ULONGLONG nAttributes,ULONGLONG nCoding)
          CONST
virtual INT GetTMParameterCodingLength() CONST
virtual ULONGLONG TranslateTMParameterCoding(LPCTSTR pszCoding,BOOL bDBMS=TRUE)
          CONST
virtual CString TranslateTMParameterCoding(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTMParameterInterpretations(CStringArray
          &szInterpretations,CLongUIntArray &nInterpretations) CONST
virtual BOOL CheckTMParameterInterpretation(ULONGLONG nAttributes,ULONGLONG
          nInterpretation) CONST
virtual INT GetTMParameterInterpretationLength() CONST
```

```
virtual ULONGLONG TranslateTMParameterInterpretation(LPCTSTR
        pszInterpretation,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMParameterInterpretation(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTMParameterValidityParameters(CStringArray &szParameters) CONST
virtual BOOL CheckTMParameterValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTMParameterValidityParameterLength() CONST
virtual BOOL CheckTMParameterValidityValue(INT nValue) CONST
virtual BOOL GetTMParameterValidityValueRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMParameterRelatedParameter(LPCTSTR pszTag) CONST
virtual BOOL CheckTMParameterRelatedParameter(ULONGLONG nAttributes,LPCTSTR
        pszTag) CONST
virtual INT GetTMParameterRelatedParameterLength() CONST
virtual CString TranslateTMParameterRelatedParameter(LPCTSTR pszTag) CONST
virtual BOOL CheckTMParameterConstValue(ULONGLONG nAttributes,LPCTSTR
        pszConstValue) CONST
virtual INT GetTMParameterConstValueLength() CONST
virtual ULONGLONG TranslateTMParameterConstValue(ULONGLONG nAttributes,INT
        nWidth,LPCTSTR pszConstValue) CONST
virtual CString TranslateTMParameterConstValue(ULONGLONG nAttributes,INT
        nWidth,ULONGLONG nConstValue) CONST
virtual BOOL CheckTMParameterDerivationExpression(LPCTSTR pszExpression) CONST
virtual INT GetTMParameterDerivationExpressionLength() CONST
virtual BOOL TranslateTMParameterDerivationExpression(LPCTSTR pszTag,LPCTSTR
        pszExpression,CONST CStringArray &pParameters,CString
        &szProcedure,CStringArray &szErrors) CONST
virtual INT EnumTMParameterTriggerTypes(CStringArray
        &szTriggerTypes,CLongUIntArray &nTriggerTypes) CONST
virtual BOOL CheckTMParameterTriggerType(ULONGLONG nAttributes,ULONGLONG
        nTriggerType) CONST
virtual INT GetTMParameterTriggerTypeLength() CONST
virtual ULONGLONG TranslateTMParameterTriggerType(LPCTSTR pszTriggerType,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMParameterTriggerType(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckTMParameterTriggerExpression(ULONGLONG nNature,ULONGLONG
        nTriggerType,LPCTSTR pszExpression) CONST
virtual INT GetTMParameterTriggerExpressionLength() CONST
virtual CString TranslateTMParameterTriggerExpression(LPCTSTR pszExpression)
        CONST
virtual BOOL TranslateTMParameterTriggerExpression(LPCTSTR pszExpression,CString
        &szParameter,CString &szPacket) CONST
virtual BOOL CheckTMParameterCalibrationTable(ULONGLONG nAttributes,LPCTSTR
        pszTable) CONST
virtual INT GetTMParameterCalibrationTableLength() CONST
```
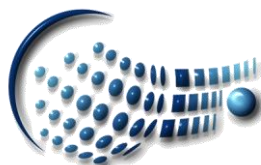
```
virtual INT GetTMParameterNumCalTablesCount() CONST
virtual INT GetTMParameterTxtCalTablesCount() CONST
virtual INT GetTMParameterNumOolTablesCount() CONST
virtual INT GetTMParameterTxtOolTablesCount() CONST
virtual BOOL CheckTMParameterSourceSubSystem(LPCTSTR pszSubSystem) CONST
virtual INT GetTMParameterSourceSubSystemLength() CONST
virtual BOOL CheckTMParameterSourceEquipment(LPCTSTR pszEquipment) CONST
virtual INT GetTMParameterSourceEquipmentLength() CONST
virtual BOOL CheckTMParameterRouterSubSystem(LPCTSTR pszSubSystem) CONST
virtual INT GetTMParameterRouterSubSystemLength() CONST
virtual BOOL CheckTMParameterRouterEquipment(LPCTSTR pszEquipment) CONST
virtual INT GetTMParameterRouterEquipmentLength() CONST
virtual BOOL CheckTMParameterOperationalInfo(LPCTSTR pszInfo) CONST
virtual INT GetTMParameterOperationalInfoLength() CONST
virtual BOOL CheckTMParameterDangerConditions(LPCTSTR pszConditions) CONST
virtual INT GetTMParameterDangerConditionsLength() CONST
virtual BOOL CheckTMParameterValidityConditions(LPCTSTR pszConditions) CONST
virtual INT GetTMParameterValidityConditionsLength() CONST
virtual BOOL CheckTMParameterAlternativeParameters(LPCTSTR pszParameters) CONST
virtual INT GetTMParameterAlternativeParametersLength() CONST
virtual BOOL CheckTMParameterAssociatedTelecommands(LPCTSTR pszTelecommands)
        CONST
virtual INT GetTMParameterAssociatedTelecommandsLength() CONST
virtual BOOL CheckTMParameterInterrogationInstruction(LPCTSTR pszInstruction)
        CONST
virtual INT GetTMParameterInterrogationInstructionLength() CONST
virtual BOOL GetTMParameterInterrogationInstructionRange(UINT &nLow,UINT &nHigh)
        CONST
virtual UINT TranslateTMParameterInterrogationInstruction(LPCTSTR
        pszInstruction) CONST
virtual CString TranslateTMParameterInterrogationInstruction(UINT nInstruction)
        CONST
virtual BOOL CheckTMParameterOnBoardID(INT nPID) CONST
virtual BOOL GetTMParameterOnBoardIDRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMParameterOnBoardTimerID(INT nOBTID) CONST
virtual BOOL GetTMParameterOnBoardTimerIDRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMParameterDecimalDigits(INT nDigits) CONST
virtual BOOL GetTMParameterDecimalDigitsRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTMParameterResultTypes(CStringArray
        &szResultTypes,CLongUIntArray &nResultTypes) CONST
virtual BOOL CheckTMParameterResultType(ULONGLONG nResultType) CONST
virtual INT GetTMParameterResultTypeLength() CONST
virtual ULONGLONG TranslateTMParameterResultType(LPCTSTR pszResultType,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMParameterResultType(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
```

```
virtual INT EnumTMParameterConsistencyChecksFlags(CStringArray
          &szFlags,CLongUIntArray &nFlags) CONST
virtual BOOL CheckTMParameterConsistencyChecksFlag(ULONGLONG nFlag) CONST
virtual INT GetTMParameterConsistenceChecksFlagLength() CONST
virtual ULONGLONG TranslateTMParameterConsistencyChecksFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMParameterConsistencyChecksFlag(ULONGLONG
          nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMParameterTimeCorrelationFlags(CStringArray
          &szFlags,CLongUIntArray &nFlags) CONST
virtual BOOL CheckTMParameterTimeCorrelationFlag(ULONGLONG nFlag) CONST
virtual INT GetTMParameterTimeCorrelationFlagLength() CONST
virtual ULONGLONG TranslateTMParameterTimeCorrelationFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMParameterTimeCorrelationFlag(ULONGLONG
          nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMParameterArchivingFlags(CStringArray &szFlags,CLongUIntArray
          &nFlags) CONST
virtual BOOL CheckTMParameterArchivingFlag(ULONGLONG nFlag) CONST
virtual INT GetTMParameterArchivingFlagLength() CONST
virtual ULONGLONG TranslateTMParameterArchivingFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMParameterArchivingFlag(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTMParameterEndianityFlags(CStringArray &szFlags,CLongUIntArray
          &nFlags) CONST
virtual BOOL CheckTMParameterEndianityFlag(ULONGLONG nFlag) CONST
virtual INT GetTMParameterEndianityFlagLength() CONST
virtual ULONGLONG TranslateTMParameterEndianityFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMParameterEndianityFlag(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual BOOL CheckTMParameterSpareList(LPCTSTR pszTag) CONST
virtual INT EnumTMParameterStatus(CStringArray &szStatus,CUIntArray &nStatus)
          CONST
virtual BOOL CheckTMParameterStatus(LPCTSTR pszStatus) CONST
virtual INT GetTMParameterStatusLength() CONST
virtual INT TranslateTMParameterStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMParameterStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

Telemetry Parameter Groups:

```
virtual BOOL CheckTMParameterGroupName(LPCTSTR pszName) CONST
virtual INT GetTMParameterGroupNameLength() CONST
virtual BOOL CheckTMParameterGroupDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMParameterGroupDescriptionLength() CONST
virtual BOOL CheckTMParameterGroupType(LPCTSTR pszType) CONST
virtual INT GetTMParameterGroupTypeLength() CONST
```

```
virtual INT TranslateTMParameterGroupType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMParameterGroupType(INT nType,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTMParameterGroupMember(LPCTSTR pszParameter) CONST
```

3.1.2.5.9. Telemetry Numerical Calibration Tables

```
virtual BOOL CheckTMNumCalTableName(LPCTSTR pszName) CONST
virtual INT GetTMNumCalTableNameLength() CONST
virtual BOOL CheckTMNumCalTableDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMNumCalTableDescriptionLength() CONST
virtual BOOL CheckTMNumCalTableUnit(LPCTSTR pszUnit) CONST
virtual INT GetTMNumCalTableUnitLength() CONST
virtual INT EnumTMNumCalTableTypes(CStringArray &szTypes,CUIntArray &nTypes)
        CONST
virtual BOOL CheckTMNumCalTableType(UINT nType) CONST
virtual INT GetTMNumCalTableTypeLength() CONST
virtual UINT TranslateTMNumCalTableType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMNumCalTableType(UINT nAttributes,BOOL bDBMS=TRUE)
        CONST
virtual INT EnumTMNumCalTableEngineeringValuesCodings(CStringArray
        &szCodings,CUIntArray &nCodings) CONST
virtual BOOL CheckTMNumCalTableEngineeringValuesCoding(UINT nCoding) CONST
virtual INT GetTMNumCalTableEngineeringValuesCodingLength() CONST
virtual UINT TranslateTMNumCalTableEngineeringValuesCoding(LPCTSTR
        pszCoding,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMNumCalTableEngineeringValuesCoding(UINT
        nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMNumCalTableRawValuesCodings(CStringArray &szCodings,CUIntArray
        &nCodings) CONST
virtual BOOL CheckTMNumCalTableRawValuesCoding(UINT nCoding) CONST
virtual INT GetTMNumCalTableRawValuesCodingLength() CONST
virtual UINT TranslateTMNumCalTableRawValuesCoding(LPCTSTR pszCoding,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMNumCalTableRawValuesCoding(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTMNumCalTableRawValuesRadixes(CStringArray &szRadixes,CUIntArray
        &nRadixes) CONST
virtual BOOL CheckTMNumCalTableRawValuesRadix(UINT nCoding,UINT nRadix) CONST
virtual INT GetTMNumCalTableRawValuesRadixLength() CONST
virtual UINT TranslateTMNumCalTableRawValuesRadix(LPCTSTR pszRadix,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMNumCalTableRawValuesRadix(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTMNumCalTableInterpretations(CStringArray
        &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTMNumCalTableInterpretation(UINT nInterpretation) CONST
```

```
virtual INT GetTMNumCalTableInterpretationLength() CONST
virtual UINT TranslateTMNumCalTableInterpretation(LPCTSTR pszInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMNumCalTableInterpretation(UINT nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual BOOL CheckTMNumCalTableValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTMNumCalTableValidityParameterLength() CONST
virtual BOOL CheckTMNumCalTableValidityValue(INT nValue) CONST
virtual BOOL GetTMNumCalTableValidityValueRange(INT &nLow,INT &nHigh) CONST
virtual BOOL GetTMNumCalTablePositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMNumCalTablePoints(INT nPoints) CONST
virtual BOOL GetTMNumCalTablePointsRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMNumCalTablePointValue(LPCTSTR pszValue) CONST
virtual BOOL CheckTMNumCalTablePointValue(UINT nAttributes,LPCTSTR pszValue)
          CONST
virtual INT GetTMNumCalTablePointValueLength() CONST
virtual double TranslateTMNumCalTablePointValue(UINT nAttributes,LPCTSTR
          pszValue) CONST
virtual CString TranslateTMNumCalTablePointValue(UINT nAttributes,double fValue)
          CONST
virtual BOOL CheckTMNumCalTablePointInterval(double X1,double X2) CONST
virtual CString GetTMNumCalTablePointOutsideText() CONST
virtual INT EnumTMNumCalTableStatus(CStringArray &szStatus,CUIntArray &nStatus)
          CONST
virtual BOOL CheckTMNumCalTableStatus(LPCTSTR pszStatus) CONST
virtual INT GetTMNumCalTableStatusLength() CONST
virtual INT TranslateTMNumCalTableStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
          CONST
virtual CString TranslateTMNumCalTableStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.10. Telemetry Textual Calibration Tables

```
virtual BOOL CheckTMTxtCalTableName(LPCTSTR pszName) CONST
virtual INT GetTMTxtCalTableNameLength() CONST
virtual BOOL CheckTMTxtCalTableDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMTxtCalTableDescriptionLength() CONST
virtual INT EnumTMTxtCalTableRawValuesCodings(CStringArray &szCodings,CUIntArray
          &nCodings) CONST
virtual BOOL CheckTMTxtCalTableRawValuesCoding(UINT nCoding) CONST
virtual INT GetTMTxtCalTableRawValuesCodingLength() CONST
virtual UINT TranslateTMTxtCalTableRawValuesCoding(LPCTSTR pszCoding,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMTxtCalTableRawValuesCoding(UINT nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual BOOL CheckTMTxtCalTableValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTMTxtCalTableValidityParameterLength() CONST
virtual BOOL CheckTMTxtCalTableValidityValue(INT nValue) CONST
virtual BOOL GetTMTxtCalTableValidityValueRange(INT &nLow,INT &nHigh) CONST
```

```
virtual BOOL GetTMTxtCalTablePositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMTxtCalTablePoints(INT nPoints) CONST
virtual BOOL GetTMTxtCalTablePointsRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMTxtCalTablePointValue(LPCTSTR pszValue) CONST
virtual BOOL CheckTMTxtCalTablePointValue(UINT nAttributes,LPCTSTR pszValue)
          CONST
virtual INT GetTMTxtCalTablePointValueLength() CONST
virtual double TranslateTMTxtCalTablePointValue(UINT nAttributes,LPCTSTR
          pszValue) CONST
virtual CString TranslateTMTxtCalTablePointValue(UINT nAttributes,double fValue)
          CONST
virtual BOOL CheckTMTxtCalTablePointInterval(double X1,double X2) CONST
virtual BOOL CheckTMTxtCalTablePointText(LPCTSTR pszText) CONST
virtual INT GetTMTxtCalTablePointTextLength() CONST
virtual CString GetTMTxtCalTablePointOutsideText() CONST
virtual INT EnumTMTxtCalTableStatus(CStringArray &szStatus,CUIntArray &nStatus)
          CONST
virtual BOOL CheckTMTxtCalTableStatus(LPCTSTR pszStatus) CONST
virtual INT GetTMTxtCalTableStatusLength() CONST
virtual INT TranslateTMTxtCalTableStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
          CONST
virtual CString TranslateTMTxtCalTableStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.11. Telemetry Numerical Out-of-Limit Tables

```
virtual BOOL CheckTMNumOolTableName(LPCTSTR pszName) CONST
virtual INT GetTMNumOolTableNameLength() CONST
virtual BOOL CheckTMNumOolTableDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMNumOolTableDescriptionLength() CONST
virtual BOOL CheckTMNumOolTableUnit(LPCTSTR pszUnit) CONST
virtual INT GetTMNumOolTableUnitLength() CONST
virtual BOOL CheckTMNumOolTableCheckCount(INT nCount) CONST
virtual BOOL GetTMNumOolTableCheckCountRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMNumOolTableCheckCount(LPCTSTR pszCount) CONST
virtual CString TranslateTMNumOolTableCheckCount(INT nCount) CONST
virtual BOOL CheckTMNumOolTableLimitCount(INT nCount) CONST
virtual BOOL GetTMNumOolTableLimitCountRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMNumOolTableLimitCount(LPCTSTR pszCount) CONST
virtual CString TranslateTMNumOolTableLimitCount(INT nCount) CONST
virtual INT EnumTMNumOolTableInterpretations(CStringArray
          &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTMNumOolTableInterpretation(UINT nInterpretation) CONST
virtual INT GetTMNumOolTableInterpretationLength() CONST
virtual UINT TranslateTMNumOolTableInterpretation(LPCTSTR pszInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMNumOolTableInterpretation(UINT nInterpretation,BOOL
          bDBMS=TRUE) CONST
```

```
virtual INT EnumTMNumOolTableCodings(CStringArray &szCodings,CUIntArray
        &nCodings) CONST
virtual BOOL CheckTMNumOolTableCoding(UINT nInterpretation,UINT nCoding) CONST
virtual INT GetTMNumOolTableCodingLength() CONST
virtual UINT TranslateTMNumOolTableCoding(UINT nInterpretation,LPCTSTR
        pszCoding,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMNumOolTableCoding(UINT nCoding,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMNumOolTableRadixes(CStringArray &szRadixes,CUIntArray
        &nRadixes) CONST
virtual BOOL CheckTMNumOolTableRadix(UINT nInterpretation,UINT nCoding,UINT
        nRadix) CONST
virtual INT GetTMNumOolTableRadixLength() CONST
virtual UINT TranslateTMNumOolTableRadix(UINT nInterpretation,LPCTSTR
        pszRadix,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMNumOolTableRadix(UINT nRadix,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMNumOolTableLimitTypes(CStringArray &szLimitTypes,CUIntArray
        &nLimitTypes) CONST
virtual BOOL CheckTMNumOolTableLimitType(UINT nLimitType) CONST
virtual INT GetTMNumOolTableLimitTypeLength() CONST
virtual INT TranslateTMNumOolTableLimitType(LPCTSTR pszLimitType,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTMNumOolTableLimitType(INT nLimitType,BOOL bDBMS=TRUE)
        CONST
virtual BOOL GetTMNumOolTableLimitPositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMNumOolTableLimitOrder(INT nOrder) CONST
virtual INT GetTMNumOolTableLimitCount(UINT nAttributes) CONST
virtual BOOL CheckTMNumOolTableLimitValue(UINT nAttributes,LPCTSTR pszLimit)
        CONST
virtual INT GetTMNumOolTableLimitValueLength() CONST
virtual double TranslateTMNumOolTableLimitValue(UINT nAttributes,LPCTSTR
        pszLimit) CONST
virtual CString TranslateTMNumOolTableLimitValue(UINT nAttributes,double fLimit)
        CONST
virtual BOOL CheckTMNumOolTableLimitValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTMNumOolTableLimitValidityParameterLength() CONST
virtual BOOL CheckTMNumOolTableLimitValidityValue(INT nValue) CONST
virtual BOOL GetTMNumOolTableLimitValidityValueRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTMNumOolTableStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
virtual BOOL CheckTMNumOolTableStatus(LPCTSTR pszStatus) CONST
virtual INT GetTMNumOolTableStatusLength() CONST
virtual INT TranslateTMNumOolTableStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTMNumOolTableStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.12. Telemetry Texual Out-of-Limit Tables

```
virtual BOOL CheckTMTxtOolTableName(LPCTSTR pszName) CONST
virtual INT GetTMTxtOolTableNameLength() CONST
virtual BOOL CheckTMTxtOolTableDescription(LPCTSTR pszDescription) CONST
virtual INT GetTMTxtOolTableDescriptionLength() CONST
virtual BOOL CheckTMTxtOolTableCheckCount(INT nCount) CONST
virtual BOOL GetTMTxtOolTableCheckCountRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTMTxtOolTableLimitCount(INT nCount) CONST
virtual BOOL GetTMTxtOolTableLimitCountRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTMTxtOolTableLimitCount(LPCTSTR pszCount) CONST
virtual CString TranslateTMTxtOolTableLimitCount(INT nCount) CONST
virtual INT TranslateTMTxtOolTableCheckCount(LPCTSTR pszCount) CONST
virtual CString TranslateTMTxtOolTableCheckCount(INT nCount) CONST
virtual INT EnumTMTxtOolTableInterpretations(CStringArray
          &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTMTxtOolTableInterpretation(UINT nInterpretation) CONST
virtual INT GetTMTxtOolTableInterpretationLength() CONST
virtual UINT TranslateTMTxtOolTableInterpretation(LPCTSTR pszInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMTxtOolTableInterpretation(UINT nInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTMTxtOolTableCodings(CStringArray &szCodings,CUIntArray
          &nCodings) CONST
virtual BOOL CheckTMTxtOolTableCoding(UINT nInterpretation,UINT nCoding) CONST
virtual INT GetTMTxtOolTableCodingLength() CONST
virtual UINT TranslateTMTxtOolTableCoding(UINT nInterpretation,LPCTSTR
          pszCoding,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMTxtOolTableCoding(UINT nCoding,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMTxtOolTableRadixes(CStringArray &szRadixes,CUIntArray
          &nRadixes) CONST
virtual BOOL CheckTMTxtOolTableRadix(UINT nInterpretation,UINT nCoding,UINT
          nRadix) CONST
virtual INT GetTMTxtOolTableRadixLength() CONST
virtual UINT TranslateTMTxtOolTableRadix(UINT nInterpretation,LPCTSTR
          pszRadix,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTMTxtOolTableRadix(UINT nRadix,BOOL bDBMS=TRUE) CONST
virtual INT EnumTMTxtOolTableLimitTypes(CStringArray &szLimitTypes,CUIntArray
          &nLimitTypes) CONST
virtual BOOL CheckTMTxtOolTableLimitType(UINT nLimitType) CONST
virtual INT GetTMTxtOolTableLimitTypeLength() CONST
virtual INT TranslateTMTxtOolTableLimitType(LPCTSTR pszLimitType,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTMTxtOolTableLimitType(INT nLimitType,BOOL bDBMS=TRUE)
          CONST
virtual BOOL GetTMTxtOolTableLimitPositionRange(INT &nLow,INT &nHigh) CONST
```

```
virtual BOOL CheckTMTxtOolTableLimitOrder(INT nOrder) CONST
virtual INT GetTMTxtOolTableLimitCount(UINT nAttributes) CONST
virtual BOOL CheckTMTxtOolTableLimitValue(UINT nAttributes,LPCTSTR pszLimit)
        CONST
virtual INT GetTMTxtOolTableLimitValueLength() CONST
virtual double TranslateTMTxtOolTableLimitValue(UINT nAttributes,LPCTSTR
        pszLimit) CONST
virtual CString TranslateTMTxtOolTableLimitValue(UINT nAttributes,double fLimit)
        CONST
virtual BOOL CheckTMTxtOolTableLimitValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTMTxtOolTableLimitValidityParameterLength() CONST
virtual BOOL CheckTMTxtOolTableLimitValidityValue(INT nValue) CONST
virtual BOOL GetTMTxtOolTableLimitValidityValueRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTMTxtOolTableStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
virtual BOOL CheckTMTxtOolTableStatus(LPCTSTR pszStatus) CONST
virtual INT GetTMTxtOolTableStatusLength() CONST
virtual INT TranslateTMTxtOolTableStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTMTxtOolTableStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.13. Telecommand Parameters

```
virtual BOOL CheckTCParameterTag(LPCTSTR pszTag) CONST
virtual INT GetTCParameterTagLength() CONST
virtual BOOL CheckTCParameterDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCParameterDescriptionLength() CONST
virtual BOOL CheckTCParameterUnit(LPCTSTR pszUnit) CONST
virtual INT GetTCParameterUnitLength() CONST
virtual INT EnumTCParameterTypes(CStringArray &szTypes,CLongUIntArray &nTypes)
        CONST
virtual BOOL CheckTCParameterType(LPCTSTR pszTag,ULONGLONG nType) CONST
virtual INT GetTCParameterTypeLength() CONST
virtual ULONGLONG TranslateTCParameterType(LPCTSTR pszType,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCParameterType(ULONGLONG nAttributes,BOOL bDBMS=TRUE)
        CONST
virtual INT EnumTCParameterCategories(CStringArray &szCategories,CLongUIntArray
        &nCategories) CONST
virtual BOOL CheckTCParameterCategory(ULONGLONG nCategory) CONST
virtual INT GetTCParameterCategoryLength() CONST
virtual ULONGLONG TranslateTCParameterCategory(LPCTSTR pszCategory,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCParameterCategory(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCParameterTypeCodes(CStringArray &szPTCs,CUIntArray &nPTCs)
        CONST
```

```
virtual BOOL CheckTCParameterTypeCode(INT nPTC) CONST
virtual BOOL GetTCParameterTypeCodeRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTCParameterTypeCode(LPCTSTR pszPTC) CONST
virtual CString TranslateTCParameterTypeCode(INT nPTC,INT nPFC) CONST
virtual INT EnumTCParameterFormatCodes(LPCTSTR pszPTC,CStringArray
        &szPFCs,CUIntArray &nPFCs) CONST
virtual BOOL CheckTCParameterFormatCode(INT nPTC,INT nPFC) CONST
virtual BOOL GetTCParameterFormatCodeRange(INT nPTC,INT &nLow,INT &nHigh) CONST
virtual INT TranslateTCParameterFormatCode(LPCTSTR pszPFC) CONST
virtual CString TranslateTCParameterFormatCode(INT nPTC,INT nPFC) CONST
virtual INT EnumTCParameterDisplayFormats(CStringArray &szFormats,CLongUIntArray
        &nFormats) CONST
virtual INT EnumTCParameterDisplayFormats(ULONGLONG nAttributes,CStringArray
        &szFormats,CLongUIntArray &nFormats) CONST
virtual BOOL CheckTCParameterDisplayFormat(ULONGLONG nAttributes,ULONGLONG
        nFormat) CONST
virtual INT GetTCParameterDisplayFormatLength() CONST
virtual ULONGLONG TranslateTCParameterDisplayFormat(LPCTSTR pszFormat,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCParameterDisplayFormat(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCParameterCodings(CStringArray &szCodings,CLongUIntArray
        &nCodings) CONST
virtual BOOL CheckTCParameterCoding(ULONGLONG nAttributes,ULONGLONG nCoding)
        CONST
virtual INT GetTCParameterCodingLength() CONST
virtual ULONGLONG TranslateTCParameterCoding(LPCTSTR pszCoding,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCParameterCoding(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCParameterRadixes(CStringArray &szRadixes,CLongUIntArray
        &nRadixes) CONST
virtual BOOL CheckTCParameterRadix(ULONGLONG nAttributes,ULONGLONG nRadix) CONST
virtual INT GetTCParameterRadixLength() CONST
virtual ULONGLONG TranslateTCParameterRadix(LPCTSTR pszRadix,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCParameterRadix(ULONGLONG nAttributes,BOOL bDBMS=TRUE)
        CONST
virtual BOOL CheckTCParameterBitWidth(INT nWidth) CONST
virtual BOOL GetTCParameterBitWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCParameterCalTable(ULONGLONG nAttributes,LPCTSTR pszTable)
        CONST
virtual INT GetTCParameterCalTableLength() CONST
```
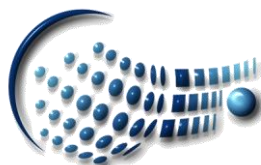
```
virtual BOOL CheckTCParameterOolTable(ULONGLONG nAttributes,LPCTSTR pszTable)
          CONST
virtual INT GetTCParameterOolTableLength() CONST
virtual INT EnumTCParameterConstValueTypes(CStringArray &szTypes,CLongUIntArray
          &nTypes) CONST
virtual INT EnumTCParameterConstValueTypes(ULONGLONG nAttributes,CStringArray
          &szTypes,CLongUIntArray &nTypes) CONST
virtual BOOL CheckTCParameterConstValueType(ULONGLONG nType,LPCTSTR
          pszConstValue) CONST
virtual INT GetTCParameterConstValueTypeLength() CONST
virtual ULONGLONG TranslateTCParameterConstValueType(LPCTSTR pszType,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCParameterConstValueType(ULONGLONG nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual BOOL CheckTCParameterConstValue(ULONGLONG nAttributes,LPCTSTR
          pszConstValue) CONST
virtual INT GetTCParameterConstValueLength() CONST
virtual ULONGLONG TranslateTCParameterConstValue(ULONGLONG nAttributes,INT
          nWidth,LPCTSTR pszConstValue) CONST
virtual CString TranslateTCParameterConstValue(ULONGLONG nAttributes,INT
          nWidth,ULONGLONG nConstValue) CONST
virtual BOOL CheckTCParameterConstTimeType(LPCTSTR pszTag) CONST
virtual BOOL CheckTCParameterConstTime(ULONGLONG nAttributes,LPCTSTR
          pszConstTime) CONST
virtual INT GetTCParameterConstTimeLength() CONST
virtual CTimeTag TranslateTCParameterConstTime(ULONGLONG nAttributes,LPCTSTR
          pszConstTime) CONST
virtual CString TranslateTCParameterConstTime(ULONGLONG nAttributes,CONST
          CTimeTag &tTime) CONST
virtual BOOL CheckTCParameterOnBoardTimerID(INT nOBTID) CONST
virtual BOOL GetTCParameterOnBoardTimerIDRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCParameterTimeCorrelationFlags(CStringArray
          &szFlags,CLongUIntArray &nFlags) CONST
virtual BOOL CheckTCParameterTimeCorrelationFlag(ULONGLONG nFlag) CONST
virtual INT GetTCParameterTimeCorrelationFlagLength() CONST
virtual ULONGLONG TranslateTCParameterTimeCorrelationFlag(LPCTSTR pszFlag,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCParameterTimeCorrelationFlag(ULONGLONG
          nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCParameterStatus(CStringArray &szStatus,CUIntArray &nStatus)
          CONST
virtual BOOL CheckTCParameterStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCParameterStatusLength() CONST
virtual INT TranslateTCParameterStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCParameterStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.14. Telecommand Numerical Calibration Tables

```
virtual BOOL CheckTCNumCalTableName(LPCTSTR pszName) CONST
virtual INT GetTCNumCalTableNameLength() CONST
virtual BOOL CheckTCNumCalTableDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCNumCalTableDescriptionLength() CONST
virtual BOOL CheckTCNumCalTableUnit(LPCTSTR pszUnit) CONST
virtual INT GetTCNumCalTableUnitLength() CONST
virtual INT EnumTCNumCalTableTypes(CStringArray &szTypes,CUIntArray &nTypes)
          CONST
virtual BOOL CheckTCNumCalTableType(UINT nType) CONST
virtual INT GetTCNumCalTableTypeLength() CONST
virtual UINT TranslateTCNumCalTableType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCNumCalTableType(UINT nAttributes,BOOL bDBMS=TRUE)
          CONST
virtual INT EnumTCNumCalTableEngineeringValuesCodings(CStringArray
          &szCodings,CUIntArray &nCodings) CONST
virtual BOOL CheckTCNumCalTableEngineeringValuesCoding(UINT nCoding) CONST
virtual INT GetTCNumCalTableEngineeringValuesCodingLength() CONST
virtual UINT TranslateTCNumCalTableEngineeringValuesCoding(LPCTSTR
          pszCoding,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCNumCalTableEngineeringValuesCoding(UINT
          nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCNumCalTableRawValuesCodings(CStringArray &szCodings,CUIntArray
          &nCodings) CONST
virtual BOOL CheckTCNumCalTableRawValuesCoding(UINT nCoding) CONST
virtual INT GetTCNumCalTableRawValuesCodingLength() CONST
virtual UINT TranslateTCNumCalTableRawValuesCoding(LPCTSTR pszCoding,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCNumCalTableRawValuesCoding(UINT nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCNumCalTableRawValuesRadixes(CStringArray &szRadixes,CUIntArray
          &nRadixes) CONST
virtual BOOL CheckTCNumCalTableRawValuesRadix(UINT nCoding,UINT nRadix) CONST
virtual INT GetTCNumCalTableRawValuesRadixLength() CONST
virtual UINT TranslateTCNumCalTableRawValuesRadix(LPCTSTR pszRadix,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCNumCalTableRawValuesRadix(UINT nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCNumCalTableInterpretations(CStringArray
          &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTCNumCalTableInterpretation(UINT nInterpretation) CONST
virtual INT GetTCNumCalTableInterpretationLength() CONST
virtual UINT TranslateTCNumCalTableInterpretation(LPCTSTR pszInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCNumCalTableInterpretation(UINT nAttributes,BOOL
          bDBMS=TRUE) CONST
```

```
virtual BOOL CheckTCNumCalTableValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTCNumCalTableValidityParameterLength() CONST
virtual BOOL CheckTCNumCalTableValidityValue(INT nValue) CONST
virtual BOOL GetTCNumCalTableValidityValueRange(INT &nLow,INT &nHigh) CONST
virtual BOOL GetTCNumCalTablePositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCNumCalTablePoints(INT nPoints) CONST
virtual BOOL GetTCNumCalTablePointsRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCNumCalTablePointValue(LPCTSTR pszValue) CONST
virtual BOOL CheckTCNumCalTablePointValue(UINT nAttributes,LPCTSTR pszValue)
        CONST
virtual INT GetTCNumCalTablePointValueLength() CONST
virtual double TranslateTCNumCalTablePointValue(UINT nAttributes,LPCTSTR
        pszValue) CONST
virtual CString TranslateTCNumCalTablePointValue(UINT nAttributes,double fValue)
        CONST
virtual BOOL CheckTCNumCalTablePointInterval(double X1,double X2) CONST
virtual CString GetTCNumCalTablePointOutsideText() CONST
virtual INT EnumTCNumCalTableStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
virtual BOOL CheckTCNumCalTableStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCNumCalTableStatusLength() CONST
virtual INT TranslateTCNumCalTableStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCNumCalTableStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.15. Telecommand Textual Calibration Tables

```
virtual BOOL CheckTCTxtCalTableName(LPCTSTR pszName) CONST
virtual INT GetTCTxtCalTableNameLength() CONST
virtual BOOL CheckTCTxtCalTableDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCTxtCalTableDescriptionLength() CONST
virtual INT EnumTCTxtCalTableRawValuesCodings(CStringArray &szCodings,CUIntArray
        &nCodings) CONST
virtual BOOL CheckTCTxtCalTableRawValuesCoding(UINT nCoding) CONST
virtual INT GetTCTxtCalTableRawValuesCodingLength() CONST
virtual UINT TranslateTCTxtCalTableRawValuesCoding(LPCTSTR pszCoding,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCTxtCalTableRawValuesCoding(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckTCTxtCalTableValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTCTxtCalTableValidityParameterLength() CONST
virtual BOOL CheckTCTxtCalTableValidityValue(INT nValue) CONST
virtual BOOL GetTCTxtCalTableValidityValueRange(INT &nLow,INT &nHigh) CONST
virtual BOOL GetTCTxtCalTablePositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCTxtCalTablePoints(INT nPoints) CONST
virtual BOOL GetTCTxtCalTablePointsRange(INT &nLow,INT &nHigh) CONST
```

```
virtual BOOL CheckTCTxtCalTablePointValue(LPCTSTR pszValue) CONST
virtual BOOL CheckTCTxtCalTablePointValue(UINT nAttributes,LPCTSTR pszValue)
          CONST
virtual INT GetTCTxtCalTablePointValueLength() CONST
virtual double TranslateTCTxtCalTablePointValue(UINT nAttributes,LPCTSTR
          pszValue) CONST
virtual CString TranslateTCTxtCalTablePointValue(UINT nAttributes,double fValue)
          CONST
virtual BOOL CheckTCTxtCalTablePointInterval(double X1,double X2) CONST
virtual BOOL CheckTCTxtCalTablePointText(LPCTSTR pszText) CONST
virtual INT GetTCTxtCalTablePointTextLength() CONST
virtual CString GetTCTxtCalTablePointOutsideText() CONST
virtual INT EnumTCTxtCalTableStatus(CStringArray &szStatus,CUIntArray &nStatus)
          CONST
virtual BOOL CheckTCTxtCalTableStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCTxtCalTableStatusLength() CONST
virtual INT TranslateTCTxtCalTableStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
          CONST
virtual CString TranslateTCTxtCalTableStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.16. Telecommand Numerical Out-of-Limit Tables

```
virtual BOOL CheckTCNumOolTableName(LPCTSTR pszName) CONST
virtual INT GetTCNumOolTableNameLength() CONST
virtual BOOL CheckTCNumOolTableDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCNumOolTableDescriptionLength() CONST
virtual BOOL CheckTCNumOolTableUnit(LPCTSTR pszUnit) CONST
virtual INT GetTCNumOolTableUnitLength() CONST
virtual BOOL CheckTCNumOolTableCheckCount(INT nCount) CONST
virtual BOOL GetTCNumOolTableCheckCountRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTCNumOolTableCheckCount(LPCTSTR pszCount) CONST
virtual CString TranslateTCNumOolTableCheckCount(INT nCount) CONST
virtual BOOL CheckTCNumOolTableLimitCount(INT nCount) CONST
virtual BOOL GetTCNumOolTableLimitCountRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTCNumOolTableLimitCount(LPCTSTR pszCount) CONST
virtual CString TranslateTCNumOolTableLimitCount(INT nCount) CONST
virtual INT EnumTCNumOolTableInterpretations(CStringArray
          &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTCNumOolTableInterpretation(UINT nInterpretation) CONST
virtual INT GetTCNumOolTableInterpretationLength() CONST
virtual UINT TranslateTCNumOolTableInterpretation(LPCTSTR pszInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCNumOolTableInterpretation(UINT nInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCNumOolTableCodings(CStringArray &szCodings,CUIntArray
          &nCodings) CONST
virtual BOOL CheckTCNumOolTableCoding(UINT nInterpretation,UINT nCoding) CONST
virtual INT GetTCNumOolTableCodingLength() CONST
```

```
virtual UINT TranslateTCNumOolTableCoding(UINT nInterpretation,LPCTSTR
        pszCoding,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCNumOolTableCoding(UINT nCoding,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCNumOolTableRadixes(CStringArray &szRadixes,CUIntArray
        &nRadixes) CONST
virtual BOOL CheckTCNumOolTableRadix(UINT nInterpretation,UINT nCoding,UINT
        nRadix) CONST
virtual INT GetTCNumOolTableRadixLength() CONST
virtual UINT TranslateTCNumOolTableRadix(UINT nInterpretation,LPCTSTR
        pszRadix,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCNumOolTableRadix(UINT nRadix,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCNumOolTableLimitTypes(CStringArray &szLimitTypes,CUIntArray
        &nLimitTypes) CONST
virtual BOOL CheckTCNumOolTableLimitType(UINT nLimitType) CONST
virtual INT GetTCNumOolTableLimitTypeLength() CONST
virtual INT TranslateTCNumOolTableLimitType(LPCTSTR pszLimitType,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCNumOolTableLimitType(INT nLimitType,BOOL bDBMS=TRUE)
        CONST
virtual BOOL GetTCNumOolTableLimitPositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCNumOolTableLimitOrder(INT nOrder) CONST
virtual INT GetTCNumOolTableLimitCount(UINT nAttributes) CONST
virtual BOOL CheckTCNumOolTableLimitValue(UINT nAttributes,LPCTSTR pszLimit)
        CONST
virtual INT GetTCNumOolTableLimitValueLength() CONST
virtual double TranslateTCNumOolTableLimitValue(UINT nAttributes,LPCTSTR
        pszLimit) CONST
virtual CString TranslateTCNumOolTableLimitValue(UINT nAttributes,double fLimit)
        CONST
virtual BOOL CheckTCNumOolTableLimitTime(ULONGLONG nAttributes,LPCTSTR pszTime)
        CONST
virtual CTimeTag TranslateTCNumOolTableLimitTime(ULONGLONG nAttributes,LPCTSTR
        pszTime) CONST
virtual CString TranslateTCNumOolTableLimitTime(ULONGLONG nAttributes,CONST
        CTimeTag &tTime) CONST
virtual BOOL CheckTCNumOolTableLimitValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTCNumOolTableLimitValidityParameterLength() CONST
virtual BOOL CheckTCNumOolTableLimitValidityValue(INT nValue) CONST
virtual BOOL GetTCNumOolTableLimitValidityValueRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCNumOolTableStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
virtual BOOL CheckTCNumOolTableStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCNumOolTableStatusLength() CONST
virtual INT TranslateTCNumOolTableStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCNumOolTableStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.17. Telecommand Texual Out-of-Limit Tables

```
virtual BOOL CheckTCTxtOolTableName(LPCTSTR pszName) CONST
virtual INT GetTCTxtOolTableNameLength() CONST
virtual BOOL CheckTCTxtOolTableDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCTxtOolTableDescriptionLength() CONST
virtual BOOL CheckTCTxtOolTableCheckCount(INT nCount) CONST
virtual BOOL GetTCTxtOolTableCheckCountRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCTxtOolTableLimitCount(INT nCount) CONST
virtual BOOL GetTCTxtOolTableLimitCountRange(INT &nLow,INT &nHigh) CONST
virtual INT TranslateTCTxtOolTableLimitCount(LPCTSTR pszCount) CONST
virtual CString TranslateTCTxtOolTableLimitCount(INT nCount) CONST
virtual INT TranslateTCTxtOolTableCheckCount(UINT nInterpretation,LPCTSTR
          pszCount) CONST
virtual CString TranslateTCTxtOolTableCheckCount(UINT nInterpretation,INT
          nCount) CONST
virtual INT EnumTCTxtOolTableInterpretations(CStringArray
          &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTCTxtOolTableInterpretation(UINT nInterpretation) CONST
virtual INT GetTCTxtOolTableInterpretationLength() CONST
virtual UINT TranslateTCTxtOolTableInterpretation(LPCTSTR pszInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCTxtOolTableInterpretation(UINT nInterpretation,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCTxtOolTableCodings(CStringArray &szCodings,CUIntArray
          &nCodings) CONST
virtual BOOL CheckTCTxtOolTableCoding(UINT nInterpretation,UINT nCoding) CONST
virtual INT GetTCTxtOolTableCodingLength() CONST
virtual UINT TranslateTCTxtOolTableCoding(UINT nInterpretation,LPCTSTR
          pszCoding,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCTxtOolTableCoding(UINT nCoding,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCTxtOolTableRadixes(CStringArray &szRadixes,CUIntArray
          &nRadixes) CONST
virtual BOOL CheckTCTxtOolTableRadix(UINT nInterpretation,UINT nCoding,UINT
          nRadix) CONST
virtual INT GetTCTxtOolTableRadixLength() CONST
virtual UINT TranslateTCTxtOolTableRadix(UINT nInterpretation,LPCTSTR
          pszRadix,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCTxtOolTableRadix(UINT nRadix,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCTxtOolTableLimitTypes(CStringArray &szLimitTypes,CUIntArray
          &nLimitTypes) CONST
virtual BOOL CheckTCTxtOolTableLimitType(UINT nLimitType) CONST
virtual INT GetTCTxtOolTableLimitTypeLength() CONST
virtual INT TranslateTCTxtOolTableLimitType(LPCTSTR pszLimitType,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCTxtOolTableLimitType(INT nLimitType,BOOL bDBMS=TRUE)
          CONST
```

```
virtual BOOL GetTCTxtOolTableLimitPositionRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckTCTxtOolTableLimitOrder(INT nOrder) CONST
virtual INT GetTCTxtOolTableLimitCount(UINT nAttributes) CONST
virtual BOOL CheckTCTxtOolTableLimitValue(UINT nAttributes,LPCTSTR pszLimit)
        CONST
virtual INT GetTCTxtOolTableLimitValueLength() CONST
virtual double TranslateTCTxtOolTableLimitValue(UINT nAttributes,LPCTSTR
        pszLimit) CONST
virtual CString TranslateTCTxtOolTableLimitValue(UINT nAttributes,double fLimit)
        CONST
virtual BOOL CheckTCTxtOolTableLimitValidityParameter(LPCTSTR pszTag) CONST
virtual INT GetTCTxtOolTableLimitValidityParameterLength() CONST
virtual BOOL CheckTCTxtOolTableLimitValidityValue(INT nValue) CONST
virtual BOOL GetTCTxtOolTableLimitValidityValueRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCTxtOolTableStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
virtual BOOL CheckTCTxtOolTableStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCTxtOolTableStatusLength() CONST
virtual INT TranslateTCTxtOolTableStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCTxtOolTableStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

3.1.2.5.18. Telecommand Pre-Execution Validation Parameter Groups

```
virtual BOOL CheckTCPreExeGroupID(LPCTSTR pszID) CONST
virtual INT GetTCPreExeGroupIDLength() CONST
virtual BOOL CheckTCPreExeGroupDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCPreExeGroupDescriptionLength() CONST
virtual INT EnumTCPreExeGroupStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
virtual BOOL CheckTCPreExeGroupStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCPreExeGroupStatusLength() CONST
virtual INT TranslateTCPreExeGroupStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCPreExeGroupStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
Validation Group Parameters:
virtual BOOL CheckTCPreExeGroupParameterTag(LPCTSTR pszTag) CONST
virtual INT GetTCPreExeGroupParameterTagLength() CONST
virtual INT EnumTCPreExeGroupParameterInterpretations(CStringArray
        &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTCPreExeGroupParameterInterpretation(UINT nInterpretation)
        CONST
virtual INT GetTCPreExeGroupParameterInterpretationLength() CONST
virtual UINT TranslateTCPreExeGroupParameterInterpretation(LPCTSTR
        pszInterpretation,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCPreExeGroupParameterInterpretation(UINT
        nAttributes,BOOL bDBMS=TRUE) CONST
```

```
virtual BOOL CheckTCPreExeGroupParameterConstValue(ULONGLONG nAttributes,LPCTSTR
          pszValue) CONST
virtual INT GetTCPreExeGroupParameterConstValueLength() CONST
virtual double TranslateTCPreExeGroupParameterConstValue(ULONGLONG
          nAttributes,LPCTSTR pszValue) CONST
virtual CString TranslateTCPreExeGroupParameterConstValue(ULONGLONG
          nAttributes,double fValue) CONST
```

3.1.2.5.19. Telecommand Execution Verification Parameter Groups

```
virtual BOOL CheckTCExeVerGroupID(LPCTSTR pszID) CONST
virtual INT GetTCExeVerGroupIDLength() CONST
virtual BOOL CheckTCExeVerGroupName(LPCTSTR pszName) CONST
virtual INT GetTCExeVerGroupNameLength() CONST
virtual BOOL CheckTCExeVerGroupDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCExeVerGroupDescriptionLength() CONST
virtual BOOL CheckTCExeVerGroupStages(INT nStages) CONST
virtual BOOL GetTCExeVerGroupStagesRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumTCExeVerGroupStatus(CStringArray &szStatus,CUIntArray &nStatus)
          CONST
virtual BOOL CheckTCExeVerGroupStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCExeVerGroupStatusLength() CONST
virtual INT TranslateTCExeVerGroupStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
          CONST
virtual CString TranslateTCExeVerGroupStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

Verification Group Parameters:

```
virtual BOOL CheckTCExeVerGroupParameterStageID(UINT nStageID) CONST
virtual BOOL GetTCExeVerGroupParameterStageIDRange(UINT &nLow,UINT &nHigh) CONST
virtual BOOL CheckTCExeVerGroupParameterStageTask(LPCTSTR pszTask) CONST
virtual INT GetTCExeVerGroupParameterStageTaskLength() CONST
virtual INT EnumTCExeVerGroupParameterStageTypes(CStringArray
          &szTypes,CUIntArray &nTypes) CONST
virtual BOOL CheckTCExeVerGroupParameterStageType(UINT nType) CONST
virtual INT GetTCExeVerGroupParameterStageTypeLength() CONST
virtual UINT TranslateTCExeVerGroupParameterStageType(LPCTSTR pszType,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCExeVerGroupParameterStageType(UINT nAttributes,BOOL
          bDBMS=TRUE) CONST
virtual INT EnumTCExeVerGroupParameterStages(CStringArray &szStageIDs,CUIntArray
          &nStageIDs) CONST
virtual BOOL CheckTCExeVerGroupParameterStage(UINT nStageID) CONST
virtual INT GetTCExeVerGroupParameterStageLength() CONST
virtual UINT TranslateTCExeVerGroupParameterStage(LPCTSTR pszStageID,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslateTCExeVerGroupParameterStage(UINT nStageID,BOOL
          bDBMS=TRUE) CONST
```

```
virtual BOOL CheckTCExeVerGroupParameterTag(LPCTSTR pszTag) CONST
virtual INT GetTCExeVerGroupParameterTagLength() CONST
virtual INT EnumTCExeVerGroupParameterTypes(CStringArray &szTypes,CUIntArray
        &nTypes) CONST
virtual BOOL CheckTCExeVerGroupParameterType(UINT nType) CONST
virtual INT GetTCExeVerGroupParameterTypeLength() CONST
virtual UINT TranslateTCExeVerGroupParameterType(LPCTSTR pszType,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCExeVerGroupParameterType(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCExeVerGroupParameterCheckings(CStringArray
        &szCheckings,CUIntArray &nCheckings) CONST
virtual BOOL CheckTCExeVerGroupParameterChecking(UINT nChecking) CONST
virtual INT GetTCExeVerGroupParameterCheckingLength() CONST
virtual UINT TranslateTCExeVerGroupParameterChecking(LPCTSTR pszChecking,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCExeVerGroupParameterChecking(UINT nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumTCExeVerGroupParameterInterpretations(CStringArray
        &szInterpretations,CUIntArray &nInterpretations) CONST
virtual BOOL CheckTCExeVerGroupParameterInterpretation(UINT nInterpretation)
        CONST
virtual INT GetTCExeVerGroupParameterInterpretationLength() CONST
virtual UINT TranslateTCExeVerGroupParameterInterpretation(LPCTSTR
        pszInterpretation,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCExeVerGroupParameterInterpretation(UINT
        nAttributes,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTCExeVerGroupParameterConstValue(ULONGLONG nAttributes,LPCTSTR
        pszValue) CONST
virtual INT GetTCExeVerGroupParameterConstValueLength() CONST
virtual double TranslateTCExeVerGroupParameterConstValue(ULONGLONG
        nAttributes,LPCTSTR pszValue) CONST
virtual CString TranslateTCExeVerGroupParameterConstValue(ULONGLONG
        nAttributes,double fValue) CONST
virtual BOOL CheckTCExeVerGroupParameterConstValueTolerance(ULONGLONG
        nAttributes,LPCTSTR pszDelta) CONST
virtual INT GetTCExeVerGroupParameterConstValueToleranceLength() CONST
virtual double TranslateTCExeVerGroupParameterConstValueTolerance(ULONGLONG
        nAttributes,LPCTSTR pszDelta) CONST
virtual CString TranslateTCExeVerGroupParameterConstValueTolerance(ULONGLONG
        nAttributes,double fDelta) CONST
virtual BOOL CheckTCExeVerGroupParameterTimeWindowOffset(CONST CTimeSpan
        &tOffset) CONST
virtual BOOL GetTCExeVerGroupParameterTimeWindowOffsetRange(UINT &nLow,UINT
        &nHigh) CONST
virtual BOOL CheckTCExeVerGroupParameterTimeWindowInterval(CONST CTimeSpan
        &tInterval) CONST
```

```
virtual BOOL GetTCExeVerGroupParameterTimeWindowIntervalRange(UINT &nLow,UINT
        &nHigh) CONST
virtual BOOL CheckTCExeVerGroupParameterTimeWindowUncertainty(CONST CTimeSpan
        &tInterval) CONST
virtual BOOL GetTCExeVerGroupParameterTimeWindowUncertaintyRange(UINT &nLow,UINT
        &nHigh) CONST
virtual BOOL CheckTCExeVerGroupParameterTimeWindowClosurePacketID(UINT nID)
        CONST
virtual BOOL GetTCExeVerGroupParameterTimeWindowClosurePacketIDRange(UINT
        &nLow,UINT &nHigh) CONST
virtual BOOL CheckTCExeVerGroupParameterEventID(LPCTSTR pszEventID) CONST
virtual INT GetTCExeVerGroupParameterEventIDLength() CONST
```

3.1.2.5.20. Telecommand Parameter Sets

```
virtual BOOL CheckTCParameterSetName(LPCTSTR pszName) CONST
virtual INT GetTCParameterSetNameLength() CONST
virtual BOOL CheckTCParameterSetDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCParameterSetDescriptionLength() CONST
virtual BOOL CheckTCParameterSetTask(LPCTSTR pszTask) CONST
virtual INT GetTCParameterSetTaskLength() CONST
virtual INT EnumTCParameterSetTypes(CStringArray &szTypes,CUIntArray &nTypes)
        CONST
virtual BOOL CheckTCParameterSetType(UINT nType) CONST
virtual INT GetTCParameterSetTypeLength() CONST
virtual UINT TranslateTCParameterSetType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCParameterSetType(UINT nAttributes,BOOL bDBMS=TRUE)
        CONST
virtual INT EnumTCParameterSetStatus(CStringArray &szStatus,CUIntArray &nStatus)
        CONST
virtual BOOL CheckTCParameterSetStatus(LPCTSTR pszStatus) CONST
virtual INT GetTCParameterSetStatusLength() CONST
virtual INT TranslateTCParameterSetStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateTCParameterSetStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckTCParameterValueSetName(LPCTSTR pszName) CONST
virtual INT GetTCParameterValueSetNameLength() CONST
virtual BOOL CheckTCParameterValueSetDescription(LPCTSTR pszDescription) CONST
virtual INT GetTCParameterValueSetDescriptionLength() CONST
```

Parameter Set Items:
```
virtual BOOL CheckTCParameterSetItemName(LPCTSTR pszName) CONST
virtual INT GetTCParameterSetItemNameLength() CONST
virtual INT EnumTCParameterSetItemValueInterpretations(CStringArray
        &szInterpretations,CLongUIntArray &nInterpretations) CONST
virtual BOOL CheckTCParameterSetItemValueInterpretation(ULONGLONG
        nAttributes,ULONGLONG nInterpretation) CONST
virtual INT GetTCParameterSetItemValueInterpretationLength() CONST
```

```
virtual ULONGLONG TranslateTCParameterSetItemValueInterpretation(LPCTSTR
        pszInterpretation,BOOL bDBMS=TRUE) CONST
virtual CString TranslateTCParameterSetItemValueInterpretation(ULONGLONG
        nAttributes,BOOL bDBMS=TRUE) CONST
virtual INT EnumTCParameterSetItemValueRadixes(CStringArray
        &szRadixes,CLongUIntArray &nRadixes) CONST
virtual BOOL CheckTCParameterSetItemValueRadix(ULONGLONG nAttributes,ULONGLONG
        nRadix) CONST
virtual INT GetTCParameterSetItemValueRadixLength() CONST
virtual ULONGLONG TranslateTCParameterSetItemValueRadix(LPCTSTR pszRadix,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateTCParameterSetItemValueRadix(ULONGLONG nAttributes,BOOL
        bDBMS=TRUE) CONST
virtual BOOL CheckTCParameterSetItemValueBitOffset(INT nOffset) CONST
virtual BOOL GetTCParameterSetItemValueBitOffsetRange(INT &nLow,INT &nHigh)
        CONST
virtual BOOL CheckTCParameterSetItemConstValue(ULONGLONG nAttributes,LPCTSTR
        pszConstValue) CONST
virtual INT GetTCParameterSetItemConstValueLength() CONST
virtual ULONGLONG TranslateTCParameterSetItemConstValue(ULONGLONG
        nAttributes,INT nWidth,LPCTSTR pszConstValue) CONST
virtual CString TranslateTCParameterSetItemConstValue(ULONGLONG nAttributes,INT
        nWidth,ULONGLONG nConstValue) CONST
virtual BOOL CheckTCParameterSetItemConstTimeType(LPCTSTR pszName) CONST
virtual BOOL CheckTCParameterSetItemConstTime(ULONGLONG nAttributes,LPCTSTR
        pszConstTime) CONST
virtual INT GetTCParameterSetItemConstTimeLength() CONST
virtual CTimeTag TranslateTCParameterSetItemConstTime(ULONGLONG
        nAttributes,LPCTSTR pszConstTime) CONST
```

3.1.2.5.21. Alphanumeric Displays

```
virtual BOOL CheckANDName(LPCTSTR pszName) CONST
virtual INT GetANDNameLength() CONST
virtual BOOL CheckANDTitle(LPCTSTR pszTitle) CONST
virtual INT GetANDTitleLength() CONST
virtual INT EnumANDColumns(CStringArray &szColumns,CUIntArray &nColumns) CONST
virtual BOOL CheckANDColumns(LPCTSTR pszColumns) CONST
virtual INT GetANDColumnsLength() CONST
virtual UINT TranslateANDColumns(LPCTSTR pszColumns,BOOL bDBMS=TRUE) CONST
virtual BOOL TranslateANDColumns(LPCTSTR pszColumns,CUIntArray &nColumns,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateANDColumns(INT nColumns,BOOL bDBMS=TRUE) CONST
virtual CString TranslateANDColumns(CONST CUIntArray &nColumns,BOOL bDBMS=TRUE)
        CONST
virtual UINT TranslateANDColumn(LPCTSTR pszColumn,BOOL bDBMS=TRUE) CONST
virtual CString TranslateANDColumn(UINT nColumn,BOOL bDBMS=TRUE) CONST
virtual INT EnumANDStatus(CStringArray &szStatus,CUIntArray &nStatus) CONST
virtual BOOL CheckANDStatus(LPCTSTR pszStatus) CONST
virtual INT GetANDStatusLength() CONST
virtual INT TranslateANDStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateANDStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

Display Parameters:

```
virtual BOOL CheckANDParameterTag(LPCTSTR pszTag) CONST
virtual INT GetANDParameterTagLength() CONST
virtual BOOL CheckANDParameterComment(LPCTSTR pszComment) CONST
virtual INT GetANDParameterCommentLength() CONST
virtual CString TranslateANDParameterComment(LPCTSTR pszComment,BOOL bDBMS=TRUE)
        CONST
virtual BOOL CheckANDParameterRow(INT nRow) CONST
virtual BOOL GetANDParameterRowRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckANDParameterCol(INT nCol) CONST
virtual BOOL GetANDParameterColRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumANDParameterModes(CStringArray &szModes,CLongUIntArray &nModes)
        CONST
virtual BOOL CheckANDParameterMode(LPCTSTR pszParameter,LPCTSTR pszMode) CONST
virtual INT GetANDParameterModeLength() CONST
virtual ULONGLONG TranslateANDParameterMode(LPCTSTR pszMode,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateANDParameterMode(ULONGLONG nMode,BOOL bDBMS=TRUE) CONST
virtual ULONGLONG TranslateANDParameterAttributes(ULONGLONG nMode,ULONGLONG
        nAttributes) CONST
virtual INT EnumANDParameterDisplayFlags(CStringArray &szFlags,CByteArray
        &nFlags) CONST
virtual BOOL CheckANDParameterDisplayFlag(LPCTSTR pszParameter,LPCTSTR pszFlag)
        CONST
virtual INT GetANDParameterDisplayFlagLength() CONST
```

```
virtual INT TranslateANDParameterDisplayFlag(LPCTSTR pszFlag,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslateANDParameterDisplayFlag(INT nFlag,BOOL bDBMS=TRUE)
        CONST
virtual BOOL CheckANDParameterOccurrence(LPCTSTR pszParameter,INT nOccurrence)
        CONST
virtual BOOL GetANDParameterOccurrenceRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckANDParameterColor(INT nColor) CONST
virtual BOOL CheckANDParameterDigits(LPCTSTR pszParameter,INT nDigits) CONST
virtual BOOL GetANDParameterDigitsRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumANDParameterAlignments(CStringArray &szAlignments,CUIntArray
        &nAlignments) CONST
virtual BOOL CheckANDParameterAlignment(LPCTSTR pszAlignment) CONST
virtual INT GetANDParameterAlignmentLength() CONST
virtual UINT TranslateANDParameterAlignment(LPCTSTR pszAlignment,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateANDParameterAlignment(UINT nAlignment,BOOL bDBMS=TRUE)
        CONST
```

3.1.2.5.22. Graphic Displays

```
virtual BOOL CheckGRDName(LPCTSTR pszName) CONST
virtual INT GetGRDNameLength() CONST
virtual BOOL CheckGRDTitle(LPCTSTR pszTitle) CONST
virtual INT GetGRDTitleLength() CONST
virtual INT EnumGRDColumns(CStringArray &szColumns,CUIntArray &nColumns) CONST
virtual BOOL CheckGRDColumns(INT nColumns) CONST
virtual BOOL TranslateGRDColumns(LPCTSTR pszColumns,CUIntArray &nColumns,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslateGRDColumns(CONST CUIntArray &nColumns,BOOL bDBMS=TRUE)
        CONST
virtual UINT TranslateGRDColumn(LPCTSTR pszColumn,BOOL bDBMS=TRUE) CONST
virtual CString TranslateGRDColumn(UINT nColumn,BOOL bDBMS=TRUE) CONST
virtual INT EnumGRDTypes(CStringArray &szTypes,CUIntArray &nTypes) CONST
virtual BOOL CheckGRDType(LPCTSTR pszType) CONST
virtual INT GetGRDTypeLength() CONST
virtual INT TranslateGRDType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateGRDType(INT nType,BOOL bDBMS=TRUE) CONST
virtual INT EnumGRDPlotTypes(CStringArray &szTypes,CUIntArray &nTypes) CONST
virtual BOOL CheckGRDPlotType(LPCTSTR pszType,LPCTSTR pszPlotType) CONST
virtual INT GetGRDPlotTypeLength() CONST
virtual INT TranslateGRDPlotType(LPCTSTR pszType,BOOL bDBMS=TRUE) CONST
virtual CString TranslateGRDPlotType(INT nType,BOOL bDBMS=TRUE) CONST
virtual INT EnumGRDPlotModes(CStringArray &szModes,CUIntArray &nModes) CONST
virtual BOOL CheckGRDPlotMode(LPCTSTR pszType,LPCTSTR pszMode) CONST
virtual INT GetGRDPlotModeLength() CONST
virtual INT TranslateGRDPlotMode(LPCTSTR pszMode,BOOL bDBMS=TRUE) CONST
virtual CString TranslateGRDPlotMode(INT nMode,BOOL bDBMS=TRUE) CONST
```

```
virtual BOOL CheckGRDPlotWidth(INT nWidth) CONST
virtual BOOL GetGRDPlotWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDPlotHeight(INT nHeight) CONST
virtual BOOL GetGRDPlotHeightRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDXGrids(LPCTSTR pszType,INT nXGrids) CONST
virtual BOOL GetGRDXGridsRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDYGrids(LPCTSTR pszType,INT nYGrids) CONST
virtual BOOL GetGRDYGridsRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDXTicks(INT nXTicks) CONST
virtual BOOL GetGRDXTicksRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDYTicks(INT nYTicks) CONST
virtual BOOL GetGRDYTicksRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDGridsWidth(INT nWidth) CONST
virtual BOOL GetGRDGridsWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDTicksWidth(INT nWidth) CONST
virtual BOOL GetGRDTicksWidthRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumGRDAxisColors(CStringArray &szColors,CUIntArray &nColors) CONST
virtual BOOL CheckGRDAxisColor(INT nColor) CONST
virtual INT GetGRDAxisColorLength() CONST
virtual COLORREF TranslateGRDAxisColor(LPCTSTR pszColor,BOOL bDBMS=TRUE) CONST
virtual CString TranslateGRDAxisColor(COLORREF nColor,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckGRDGridsColor(INT nGridsColor) CONST
virtual BOOL CheckGRDTicksColor(INT nTicksColor) CONST
virtual BOOL CheckGRDBackgroundColor(INT nBackgroundColor) CONST
virtual BOOL CheckGRDLabelColor(INT nLabelColor) CONST
virtual BOOL CheckGRDDurationDays(INT nDays,INT nHours,INT nMinutes) CONST
virtual BOOL GetGRDDurationDaysRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDDurationHours(INT nDays,INT nHours,INT nMinutes) CONST
virtual BOOL GetGRDDurationHoursRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDDurationMinutes(INT nDays,INT nHours,INT nMinutes) CONST
virtual BOOL GetGRDDurationMinutesRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDInterval(LPCTSTR pszMode,INT nDays,INT nHours,INT
         nMinutes,INT nInterval) CONST
virtual BOOL GetGRDIntervalRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDUpdateInterval(INT nDays,INT nHours,INT nMinutes,INT
         nInterval) CONST
virtual BOOL GetGRDUpdateIntervalRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumGRDHardcopyModes(CStringArray &szModes,CByteArray &nModes) CONST
virtual BOOL CheckGRDHardcopyMode(LPCTSTR pszMode) CONST
virtual INT GetGRDHardcopyModeLength() CONST
virtual INT TranslateGRDHardcopyMode(LPCTSTR pszMode,BOOL bDBMS=TRUE) CONST
virtual CString TranslateGRDHardcopyMode(INT nMode,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckGRDParameters(LPCTSTR pszType,INT nParameters) CONST
virtual BOOL GetGRDParametersRange(INT &nLow,INT &nHigh) CONST
```

```
virtual INT EnumGRDStatus(CStringArray &szStatus,CUIntArray &nStatus) CONST
virtual BOOL CheckGRDStatus(LPCTSTR pszStatus) CONST
virtual INT GetGRDStatusLength() CONST
virtual INT TranslateGRDStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslateGRDStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

Display Parameters:

```
virtual BOOL CheckGRDParameterTag(LPCTSTR pszTag) CONST
virtual INT GetGRDParameterTagLength() CONST
virtual BOOL CheckGRDParameterComment(LPCTSTR pszComment) CONST
virtual INT GetGRDParameterCommentLength() CONST
virtual INT EnumGRDParameterPositions(CStringArray &szPositions,CUIntArray
            &nPositions) CONST
virtual BOOL CheckGRDParameterPosition(INT nPosition) CONST
virtual BOOL GetGRDParameterPositionRange(INT &nLow,INT &nHigh) CONST
virtual INT GetGRDParameterPositionLength() CONST
virtual INT TranslateGRDParameterPosition(LPCTSTR pszPosition,BOOL bDBMS=TRUE)
            CONST
virtual CString TranslateGRDParameterPosition(INT nPosition,BOOL bDBMS=TRUE)
            CONST
virtual INT EnumGRDParameterModes(CStringArray &szModes,CLongUIntArray &nModes)
            CONST
virtual BOOL CheckGRDParameterMode(LPCTSTR pszMode) CONST
virtual INT GetGRDParameterModeLength() CONST
virtual ULONGLONG TranslateGRDParameterMode(LPCTSTR pszMode,BOOL bDBMS=TRUE)
            CONST
virtual CString TranslateGRDParameterMode(ULONGLONG nMode,BOOL bDBMS=TRUE) CONST
virtual ULONGLONG TranslateGRDParameterAttributes(ULONGLONG nMode,ULONGLONG
            nAttributes) CONST
virtual BOOL CheckGRDParameterOccurrence(INT nOccurrence) CONST
virtual BOOL GetGRDParameterOccurrenceRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumGRDParameterColors(CStringArray &szColors,CUIntArray &nColors)
            CONST
virtual BOOL CheckGRDParameterColor(INT nColor) CONST
virtual INT GetGRDParameterColorLength() CONST
virtual INT TranslateGRDParameterColor(LPCTSTR pszColor,BOOL bDBMS=TRUE) CONST
virtual CString TranslateGRDParameterColor(INT nColor,BOOL bDBMS=TRUE) CONST
virtual BOOL CheckGRDParameterDigits(INT nDigits) CONST
virtual BOOL GetGRDParameterDigitsRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumGRDParameterLineStyles(CStringArray &szStyles,CUIntArray
            &nStyles) CONST
virtual BOOL CheckGRDParameterLineStyle(LPCTSTR pszStyle) CONST
virtual INT GetGRDParameterLineStyleLength() CONST
virtual INT TranslateGRDParameterLineStyle(LPCTSTR pszStyle,BOOL bDBMS=TRUE)
            CONST
virtual CString TranslateGRDParameterLineStyle(INT nStyle,BOOL bDBMS=TRUE) CONST
virtual INT EnumGRDParameterLineSymbols(CStringArray &szSymbols,CUIntArray
            &nSymbols) CONST
```

```
virtual BOOL CheckGRDParameterLineSymbol(LPCTSTR pszSymbol) CONST
virtual INT GetGRDParameterLineSymbolLength() CONST
virtual INT TranslateGRDParameterLineSymbol(LPCTSTR pszSymbol,BOOL bDBMS=TRUE)
          CONST
virtual CString TranslateGRDParameterLineSymbol(INT nSymbol,BOOL bDBMS=TRUE)
          CONST
virtual BOOL CheckGRDParameterLineWidth(INT nWidth) CONST
virtual BOOL GetGRDParameterLineWidthRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckGRDParameterMinimum(LPCTSTR pszMinimum) CONST
virtual BOOL CheckGRDParameterMinimum(double fMinimum) CONST
virtual INT GetGRDParameterMinimumLength() CONST
virtual BOOL CheckGRDParameterMaximum(LPCTSTR pszMaximum) CONST
virtual BOOL CheckGRDParameterMaximum(double fMaximum) CONST
virtual INT GetGRDParameterMaximumLength() CONST
virtual BOOL CheckGRDParameterRange(double fMinimum,double fMaximum) CONST
```

3.1.2.5.23. Parameter Observation Displays

```
virtual BOOL CheckPODName(LPCTSTR pszName) CONST
virtual INT GetPODNameLength() CONST
virtual BOOL CheckPODTitle(LPCTSTR pszTitle) CONST
virtual INT GetPODTitleLength() CONST
virtual INT EnumPODColumns(CStringArray &szColumns,CUIntArray &nColumns) CONST
virtual BOOL CheckPODColumns(INT nColumns) CONST
virtual BOOL TranslatePODColumns(LPCTSTR pszColumns,CUIntArray &nColumns,BOOL
          bDBMS=TRUE) CONST
virtual CString TranslatePODColumns(CONST CUIntArray &nColumns,BOOL bDBMS=TRUE)
          CONST
virtual UINT TranslatePODColumn(LPCTSTR pszColumn,BOOL bDBMS=TRUE) CONST
virtual CString TranslatePODColumn(UINT nColumn,BOOL bDBMS=TRUE) CONST
virtual INT EnumPODForegroundColors(CStringArray &szColors,CUIntArray &nColors)
          CONST
virtual BOOL CheckPODForegroundColor(INT nColor) CONST
virtual INT GetPODForegroundColorLength() CONST
virtual INT TranslatePODForegroundColor(LPCTSTR pszColor,BOOL bDBMS=TRUE) CONST
virtual CString TranslatePODForegroundColor(INT nColor,BOOL bDBMS=TRUE) CONST
virtual INT EnumPODBackgroundColors(CStringArray &szColors,CUIntArray &nColors)
          CONST
virtual BOOL CheckPODBackgroundColor(INT nColor) CONST
virtual INT GetPODBackgroundColorLength() CONST
virtual INT TranslatePODBackgroundColor(LPCTSTR pszColor,BOOL bDBMS=TRUE) CONST
virtual CString TranslatePODBackgroundColor(INT nColor,BOOL bDBMS=TRUE) CONST
virtual INT EnumPODStatus(CStringArray &szStatus,CUIntArray &nStatus) CONST
virtual BOOL CheckPODStatus(LPCTSTR pszStatus) CONST
virtual INT GetPODStatusLength() CONST
virtual INT TranslatePODStatus(LPCTSTR pszStatus,BOOL bDBMS=TRUE) CONST
virtual CString TranslatePODStatus(INT nStatus,BOOL bDBMS=TRUE) CONST
```

```
virtual INT EnumPODExpressionStack(CStringArray &szExpressions,CUIntArray
        &nExpressions) CONST
virtual BOOL CheckPODExpressionStack(LPCTSTR pszStack) CONST
virtual INT GetPODExpressionStackLength() CONST
virtual INT TranslatePODExpressionStack(LPCTSTR pszStack,CStringArray
        &szStack,CByteArray &bFunctions,CByteArray &bOperators,CByteArray
        &bParameters,CByteArray &bNumbers) CONST
virtual CString TranslatePODExpressionStack(CONST CStringArray &szStack) CONST
virtual BOOL TranslatePODExpressionStackElement(LPCTSTR
        pszElement,CPODLayoutExpression *pElement) CONST
virtual BOOL CheckPODExpressionStackElementComment(LPCTSTR pszComment) CONST
virtual INT GetPODExpressionStackElementCommentLength() CONST
virtual BOOL CheckPODExpressionStackElementColor(INT nColor) CONST
virtual BOOL CheckPODParameters(LPCTSTR pszParameters) CONST
virtual INT GetPODParametersListLength() CONST
virtual INT GetPODParameterTagLength() CONST
virtual INT TranslatePODParameters(LPCTSTR pszParameters,CStringArray
        &szParameters) CONST
virtual CString TranslatePODParameters(CONST CStringArray &szParameters) CONST
virtual BOOL CheckPODParameterSample(INT nSample) CONST
virtual BOOL GetPODParameterSampleRange(INT &nLow,INT &nHigh) CONST
virtual BOOL CheckPODParameterPosition(INT nPosition) CONST
virtual BOOL GetPODParameterPositionRange(INT &nLow,INT &nHigh) CONST
virtual INT EnumPODParameterModes(CStringArray &szModes,CLongUIntArray &nModes)
        CONST
virtual BOOL CheckPODParameterMode(LPCTSTR pszMode) CONST
virtual INT GetPODParameterModeLength() CONST
virtual ULONGLONG TranslatePODParameterMode(LPCTSTR pszMode,BOOL bDBMS=TRUE)
        CONST
virtual CString TranslatePODParameterMode(ULONGLONG nMode,BOOL bDBMS=TRUE) CONST
virtual ULONGLONG TranslatePODParameterAttributes(ULONGLONG nMode,ULONGLONG
        nAttributes) CONST
virtual INT EnumPODParameterUpdateWhenScrollFlags(CStringArray
        &szFlags,CUIntArray &nFlags) CONST
virtual BOOL CheckPODParameterUpdateWhenScrollFlag(LPCTSTR pszFlag) CONST
virtual INT GetPODParameterUpdateWhenScrollFlagLength() CONST
virtual INT TranslatePODParameterUpdateWhenScrollFlag(LPCTSTR pszFlag,BOOL
        bDBMS=TRUE) CONST
virtual CString TranslatePODParameterUpdateWhenScrollFlag(INT nFlag,BOOL
        bDBMS=TRUE) CONST
virtual INT EnumPODParameterDisplayFlags(CStringArray &szFlags,CByteArray
        &nFlags) CONST
virtual BOOL CheckPODParameterDisplayFlag(LPCTSTR pszFlag) CONST
virtual INT GetPODParameterDisplayFlagLength() CONST
virtual BOOL TranslatePODParameterDisplayFlag(LPCTSTR pszFlag,BOOL bDBMS=TRUE)
        CONST
```

```
virtual CString TranslatePODParameterDisplayFlag(BOOL bFlag,BOOL bDBMS=TRUE)
        CONST
virtual BOOL CheckPODParametersColor(INT nColor) CONST
virtual INT EnumPODFunctions(CStringArray &szFunctions,CUIntArray &nFunctions)
        CONST
virtual BOOL CheckPODFunctions(LPCTSTR pszFunctions) CONST
virtual INT GetPODFunctionsListLength() CONST
virtual INT GetPODFunctionLength() CONST
virtual INT TranslatePODFunctions(LPCTSTR pszFunctions,CStringArray
        &szFunctions) CONST
virtual CString TranslatePODFunctions(CONST CStringArray &szFunctions) CONST
virtual BOOL CheckPODFunctionsColor(INT nColor) CONST
```

3.1.2.6. Database Defaults

There are a number of procedures that allow the specification of various database defaults related to displays like colors, column widths and more.

Note:

All widths returned are in character counts (not pixels).

The following chapters list all procedures that need to be implemented.

3.1.2.6.1. Alphanumeric Display Defaults

```
virtual INT GetANDIDColumnCharWidth() CONST
virtual COLORREF GetANDIDColumnColor() CONST
virtual INT GetANDDescriptionColumnCharWidth() CONST
virtual COLORREF GetANDDescriptionColumnColor() CONST
virtual INT GetANDValueColumnCharWidth() CONST
virtual COLORREF GetANDValueColumnColor() CONST
virtual INT GetANDUnitColumnCharWidth() CONST
virtual COLORREF GetANDUnitColumnColor() CONST
virtual INT GetANDChangeTMUnitColumnCharWidth() CONST
virtual COLORREF GetANDChangeTMUnitColumnColor() CONST
virtual INT GetANDUpdateTMUnitColumnCharWidth() CONST
virtual COLORREF GetANDUpdateTMUnitColumnColor() CONST
virtual INT GetANDChangeTimeColumnCharWidth() CONST
virtual COLORREF GetANDChangeTimeColumnColor() CONST
virtual INT GetANDUpdateTimeColumnCharWidth() CONST
virtual COLORREF GetANDUpdateTimeColumnColor() CONST
virtual INT GetANDExpectedValuesColumnCharWidth() CONST
virtual COLORREF GetANDExpectedValuesColumnColor() CONST
```

3.1.2.6.2. Graphic Display Defaults

```
virtual INT GetGRDIDColumnCharWidth() CONST
virtual COLORREF GetGRDIDColumnColor() CONST
virtual INT GetGRDDescriptionColumnCharWidth() CONST
virtual COLORREF GetGRDDescriptionColumnColor() CONST
virtual INT GetGRDValueColumnCharWidth() CONST
virtual COLORREF GetGRDValueColumnColor() CONST
virtual INT GetGRDUnitColumnCharWidth() CONST
virtual COLORREF GetGRDUnitColumnColor() CONST
virtual INT GetGRDRangeColumnCharWidth() CONST
virtual COLORREF GetGRDRangeColumnColor() CONST
virtual INT GetGRDLineColumnCharWidth() CONST
virtual INT GetGRDChangeTMUnitColumnCharWidth() CONST
virtual COLORREF GetGRDChangeTMUnitColumnColor() CONST
virtual INT GetGRDUpdateTMUnitColumnCharWidth() CONST
virtual COLORREF GetGRDUpdateTMUnitColumnColor() CONST
virtual INT GetGRDChangeTimeColumnCharWidth() CONST
virtual COLORREF GetGRDChangeTimeColumnColor() CONST
virtual INT GetGRDUpdateTimeColumnCharWidth() CONST
virtual COLORREF GetGRDUpdateTimeColumnColor() CONST
```

3.1.2.6.3. Out-of-Limit Display Defaults

```
virtual INT GetOOLIDColumnCharWidth() CONST
virtual INT GetOOLDescriptionColumnCharWidth() CONST
virtual INT GetOOLValueColumnCharWidth() CONST
virtual INT GetOOLLimitColumnCharWidth() CONST
virtual INT GetOOLUnitColumnCharWidth() CONST
virtual INT GetOOLTypeColumnCharWidth() CONST
virtual INT GetOOLTMTimeColumnCharWidth() CONST
virtual INT GetOOLTMUnitColumnCharWidth() CONST
```

3.1.2.6.4. Parameter Observation Display Defaults

```
virtual INT GetPODIDColumnCharWidth() CONST
virtual INT GetPODDescriptionColumnCharWidth() CONST
virtual INT GetPODValueColumnCharWidth() CONST
virtual INT GetPODUnitColumnCharWidth() CONST
virtual INT GetPODTimeColumnCharWidth() CONST
virtual COLORREF GetPODFunctionsColor() CONST
virtual COLORREF GetPODParametersColor() CONST
virtual COLORREF GetPODExpressionColor() CONST
```

3.1.2.6.5. Telemetry Parameter Export Protocol Display Defaults

```
virtual INT GetTPEPIDColumnCharWidth() CONST
virtual COLORREF GetTPEPIDColumnColor() CONST
virtual INT GetTPEPDescriptionColumnCharWidth() CONST
virtual COLORREF GetTPEPDescriptionColumnColor() CONST
virtual INT GetTPEPValueColumnCharWidth() CONST
virtual COLORREF GetTPEPValueColumnColor() CONST
virtual INT GetTPEPUnitColumnCharWidth() CONST
virtual COLORREF GetTPEPUnitColumnColor() CONST
virtual INT GetTPEPChangeTMUnitColumnCharWidth() CONST
virtual COLORREF GetTPEPChangeTMUnitColumnColor() CONST
virtual INT GetTPEPUpdateTMUnitColumnCharWidth() CONST
virtual COLORREF GetTPEPUpdateTMUnitColumnColor() CONST
virtual INT GetTPEPChangeTimeColumnCharWidth() CONST
virtual COLORREF GetTPEPChangeTimeColumnColor() CONST
virtual INT GetTPEPUpdateTimeColumnCharWidth() CONST
virtual COLORREF GetTPEPUpdateTimeColumnColor() CONST
```

3.1.2.6.6. Manual Telecommand Stack Defaults

```
virtual INT GetTCManualStackNumberColumnCharWidth() CONST
virtual COLORREF GetTCManualStackNumberColumnColor() CONST
virtual INT GetTCManualStackNameColumnCharWidth() CONST
virtual COLORREF GetTCManualStackNameColumnColor() CONST
virtual INT GetTCManualStackDescriptionColumnCharWidth() CONST
virtual COLORREF GetTCManualStackDescriptionColumnColor() CONST
virtual INT GetTCManualStackCategoryColumnCharWidth() CONST
virtual COLORREF GetTCManualStackCategoryColumnColor() CONST
virtual INT GetTCManualStackSubSystemColumnCharWidth() CONST
virtual COLORREF GetTCManualStackSubSystemColumnColor() CONST
virtual INT GetTCManualStackArgumentsColumnCharWidth() CONST
virtual COLORREF GetTCManualStackArgumentsColumnColor() CONST
virtual INT GetTCManualStackFlagsColumnCharWidth() CONST
virtual COLORREF GetTCManualStackFlagsColumnColor() CONST
virtual INT GetTCManualStackTCPacketColumnCharWidth() CONST
virtual COLORREF GetTCManualStackTCPacketColumnColor() CONST
virtual INT GetTCManualStackDispatchTimeColumnCharWidth() CONST
virtual COLORREF GetTCManualStackDispatchTimeColumnColor() CONST
virtual INT GetTCManualStackReleaseTimeColumnCharWidth() CONST
virtual COLORREF GetTCManualStackReleaseTimeColumnColor() CONST
virtual INT GetTCManualStackReleaseTimeOffsetColumnCharWidth() CONST
virtual COLORREF GetTCManualStackReleaseTimeOffsetColumnColor() CONST
virtual INT GetTCManualStackExecutionTimeColumnCharWidth() CONST
virtual COLORREF GetTCManualStackExcutionTimeColumnColor() CONST
virtual INT GetTCManualStackExecutionTimeOffsetColumnCharWidth() CONST
virtual COLORREF GetTCManualStackExcutionTimeOffsetColumnColor() CONST
virtual INT GetTCManualStackAuthorizationColumnCharWidth() CONST
```

```
virtual COLORREF GetTCManualStackAuthorizationColumnColor() CONST
virtual INT GetTCManualStackPTVTimeWindowColumnCharWidth() CONST
virtual COLORREF GetTCManualStackPTVTimeWindowColumnColor() CONST
virtual INT GetTCManualStackPTVChecksColumnCharWidth() CONST
virtual COLORREF GetTCManualStackPTVChecksColumnColor() CONST
virtual INT GetTCManualStackPTVResultColumnCharWidth() CONST
virtual COLORREF GetTCManualStackPTVResultColumnColor() CONST
virtual INT GetTCManualStackPEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCManualStackPEVChecksColumnColor() CONST
virtual INT GetTCManualStackPEVResultColumnCharWidth() CONST
virtual COLORREF GetTCManualStackPEVResultColumnColor() CONST
virtual INT GetTCManualStackCEVTimeWindowColumnCharWidth() CONST
virtual COLORREF GetTCManualStackCEVTimeWindowColumnColor() CONST
virtual INT GetTCManualStackCEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCManualStackCEVChecksColumnColor() CONST
virtual INT GetTCManualStackStatusColumnCharWidth() CONST
virtual COLORREF GetTCManualStackStatusColumnColor() CONST
```

3.1.2.6.7. Automatic Telecommand Stack Defaults

```
virtual INT GetTCAutomaticStackNumberColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackNumberColumnColor() CONST
virtual INT GetTCAutomaticStackNameColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackNameColumnColor() CONST
virtual INT GetTCAutomaticStackDescriptionColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackDescriptionColumnColor() CONST
virtual INT GetTCAutomaticStackCategoryColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackCategoryColumnColor() CONST
virtual INT GetTCAutomaticStackSubSystemColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackSubSystemColumnColor() CONST
virtual INT GetTCAutomaticStackArgumentsColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackArgumentsColumnColor() CONST
virtual INT GetTCAutomaticStackFlagsColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackFlagsColumnColor() CONST
virtual INT GetTCAutomaticStackTCPacketColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackTCPacketColumnColor() CONST
virtual INT GetTCAutomaticStackDispatchTimeColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackDispatchTimeColumnColor() CONST
virtual INT GetTCAutomaticStackReleaseTimeColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackReleaseTimeColumnColor() CONST
virtual INT GetTCAutomaticStackReleaseTimeOffsetColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackReleaseTimeOffsetColumnColor() CONST
virtual INT GetTCAutomaticStackTransmissionTimeColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackTransmissionTimeColumnColor() CONST
virtual INT GetTCAutomaticStackExecutionTimeColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackExcutionTimeColumnColor() CONST
virtual INT GetTCAutomaticStackExecutionTimeOffsetColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackExcutionTimeOffsetColumnColor() CONST
```

```
virtual INT GetTCAutomaticStackAuthorizationColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackAuthorizationColumnColor() CONST
virtual INT GetTCAutomaticStackPTVTimeWindowColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackPTVTimeWindowColumnColor() CONST
virtual INT GetTCAutomaticStackPTVChecksColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackPTVChecksColumnColor() CONST
virtual INT GetTCAutomaticStackPTVResultColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackPTVResultColumnColor() CONST
virtual INT GetTCAutomaticStackPEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackPEVChecksColumnColor() CONST
virtual INT GetTCAutomaticStackPEVResultColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackPEVResultColumnColor() CONST
virtual INT GetTCAutomaticStackCEVTimeWindowColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackCEVTimeWindowColumnColor() CONST
virtual INT GetTCAutomaticStackCEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackCEVChecksColumnColor() CONST
virtual INT GetTCAutomaticStackCEVResultColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackCEVResultColumnColor() CONST
virtual INT GetTCAutomaticStackStatusColumnCharWidth() CONST
virtual COLORREF GetTCAutomaticStackStatusColumnColor() CONST
```

3.1.2.6.8. Telecommand Procedures Monitor Defaults

```
virtual INT GetTCProceduresMonitorNameColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorNameColumnColor() CONST
virtual INT GetTCProceduresMonitorDescriptionColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorDescriptionColumnColor() CONST
virtual INT GetTCProceduresMonitorCommentsColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorCommentsColumnColor() CONST
virtual INT GetTCProceduresMonitorAuthorColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorAuthorColumnColor() CONST
virtual INT GetTCProceduresMonitorCreationTimeColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorCreationTimeColumnColor() CONST
virtual INT GetTCProceduresMonitorModificationTimeColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorModificationTimeColumnColor() CONST
virtual INT GetTCProceduresMonitorVerificationStatusColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorVerificationStatusColumnColor() CONST
virtual INT GetTCProceduresMonitorNumberOfThreadsColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorNumberOfThreadsColumnColor() CONST
virtual INT GetTCProceduresMonitorArgumentsColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorArgumentsColumnColor() CONST
virtual INT GetTCProceduresMonitorTriggerTypeColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorTriggerTypeColumnColor() CONST
virtual INT GetTCProceduresMonitorExecutionModeColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorExecutionModeColumnColor() CONST
virtual INT GetTCProceduresMonitorExecutionCountColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorExecutionCountColumnColor() CONST
virtual INT GetTCProceduresMonitorRunningStateColumnCharWidth() CONST
```

```
virtual COLORREF GetTCProceduresMonitorRunningStateColumnColor() CONST
virtual INT GetTCProceduresMonitorAcknowledgeRequestColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorAcknowledgeRequestColumnColor() CONST
virtual INT GetTCProceduresMonitorAckRequestTimeColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorAckRequestTimeColumnColor() CONST
virtual INT GetTCProceduresMonitorPEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorPEVChecksColumnColor() CONST
virtual INT GetTCProceduresMonitorCEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorCEVChecksColumnColor() CONST
virtual INT GetTCProceduresMonitorStartTimeColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorStartTimeColumnColor() CONST
virtual INT GetTCProceduresMonitorEndTimeColumnCharWidth() CONST
virtual COLORREF GetTCProceduresMonitorEndTimeColumnColor() CONST
```

3.1.2.6.9. On-Board Diagnostics Oversampling Dump Display Defaults

```
virtual INT GetTCDiagnosticsOversamplingDumpParametersNumberCharWidth() CONST
virtual COLORREF GetTCDiagnosticsOversamplingDumpParametersNumberColumnColor()
        CONST
virtual INT GetTCDiagnosticsOversamplingDumpParametersTimeCharWidth() CONST
virtual COLORREF GetTCDiagnosticsOversamplingDumpParametersTimeColumnColor()
        CONST
virtual INT GetTCDiagnosticsOversamplingDumpParametersValueCharWidth() CONST
virtual COLORREF GetTCDiagnosticsOversamplingDumpParametersValueColumnColor()
        CONST
virtual INT GetTCDiagnosticsOversamplingDumpParametersUnitCharWidth() CONST
virtual COLORREF GetTCDiagnosticsOversamplingDumpParametersUnitColumnColor()
        CONST
virtual INT GetTCDiagnosticsOversamplingDumpRawDataNumberCharWidth() CONST
virtual COLORREF GetTCDiagnosticsOversamplingDumpRawDataNumberColumnColor()
        CONST
virtual INT GetTCDiagnosticsOversamplingDumpRawDataTimeCharWidth() CONST
virtual COLORREF GetTCDiagnosticsOversamplingDumpRawDataTimeColumnColor() CONST
virtual INT GetTCDiagnosticsOversamplingDumpRawDataContentsCharWidth() CONST
virtual COLORREF GetTCDiagnosticsOversamplingDumpRawDataContentsColumnColor()
        CONST
```

3.1.2.6.10. On-Board Buffer Queues Display Defaults

```
virtual INT GetTCOnBoardBufferQueuesSummaryNameColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesSummaryNameColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesSummaryAPIDColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesSummaryAPIDColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesSummarySubScheduleIDColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesSummarySubScheduleIDColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesSummaryLimitColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesSummaryLimitColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesSummaryCountColumnCharWidth() CONST
```

```
virtual COLORREF GetTCOnBoardBufferQueuesSummaryCountColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesSummaryUpdateTimeColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesSummaryUpdateTimeColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesSummaryStatusColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesSummaryStatusColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsNumberColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsNumberColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsNameColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsNameColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsComputerColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsComputerColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsCommandSourceColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsCommandSourceColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsDescriptionColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsDescriptionColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsCategoryColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsCategoryColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsSubSystemColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsSubSystemColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsArgumentsColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsArgumentsColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsFlagsColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsFlagsColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsTCPacketColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsTCPacketColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsPSCColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsPSCColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsDispatchTimeColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsDispatchTimeColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsReleaseTimeColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsReleaseTimeColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsReleaseTimeOffsetColumnCharWidth()
          CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsReleaseTimeOffsetColumnColor()
          CONST
virtual INT GetTCOnBoardBufferQueuesDetailsTransmissionTimeColumnCharWidth()
          CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsTransmissionTimeColumnColor()
          CONST
virtual INT GetTCOnBoardBufferQueuesDetailsExecutionTimeColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsExcutionTimeColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsExecutionTimeOffsetColumnCharWidth()
          CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsExcutionTimeOffsetColumnColor()
          CONST
virtual INT GetTCOnBoardBufferQueuesDetailsAuthorizationColumnCharWidth() CONST
```

```
virtual COLORREF GetTCOnBoardBufferQueuesDetailsAuthorizationColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsPTVTimeWindowColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsPTVTimeWindowColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsPTVChecksColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsPTVChecksColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsPTVResultColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsPTVResultColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsPEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsPEVChecksColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsPEVResultColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsPEVResultColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsCEVTimeWindowColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsCEVTimeWindowColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsCEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsCEVChecksColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsCEVResultColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsCEVResultColumnColor() CONST
virtual INT GetTCOnBoardBufferQueuesDetailsStatusColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardBufferQueuesDetailsStatusColumnColor() CONST
```

3.1.2.6.11. On-Board Memory Images Display Defaults

```
virtual INT GetTCOnBoardMemoryImagesProcessorColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardMemoryImagesProcessorColumnColor() CONST
virtual INT GetTCOnBoardMemoryImagesAddressRangeColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardMemoryImagesAddressRangeColumnColor() CONST
virtual INT GetTCOnBoardMemoryImagesSizeColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardMemoryImagesSizeColumnColor() CONST
virtual INT GetTCOnBoardMemoryImagesPatchTCColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardMemoryImagesPatchTCColumnColor() CONST
virtual INT GetTCOnBoardMemoryImagesLastPatchTimeColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardMemoryImagesLastPatchTimeColumnColor() CONST
virtual INT GetTCOnBoardMemoryImagesDumpTCColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardMemoryImagesDumpTCColumnColor() CONST
virtual INT GetTCOnBoardMemoryImagesDumpTMColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardMemoryImagesDumpTMColumnColor() CONST
virtual INT GetTCOnBoardMemoryImagesLastDumpTimeColumnCharWidth() CONST
virtual COLORREF GetTCOnBoardMemoryImagesLastDumpTimeColumnColor() CONST
```

3.1.2.6.12. Telecommand History Display Defaults

```
virtual INT GetTCHistoryNumberColumnCharWidth() CONST
virtual COLORREF GetTCHistoryNumberColumnColor() CONST
virtual INT GetTCHistoryNameColumnCharWidth() CONST
virtual COLORREF GetTCHistoryNameColumnColor() CONST
virtual INT GetTCHistoryComputerColumnCharWidth() CONST
virtual COLORREF GetTCHistoryComputerColumnColor() CONST
virtual INT GetTCHistoryCommandSourceColumnCharWidth() CONST
```

```
virtual COLORREF GetTCHistoryCommandSourceColumnColor() CONST
virtual INT GetTCHistoryDescriptionColumnCharWidth() CONST
virtual COLORREF GetTCHistoryDescriptionColumnColor() CONST
virtual INT GetTCHistoryCategoryColumnCharWidth() CONST
virtual COLORREF GetTCHistoryCategoryColumnColor() CONST
virtual INT GetTCHistorySubSystemColumnCharWidth() CONST
virtual COLORREF GetTCHistorySubSystemColumnColor() CONST
virtual INT GetTCHistoryArgumentsColumnCharWidth() CONST
virtual COLORREF GetTCHistoryArgumentsColumnColor() CONST
virtual INT GetTCHistoryFlagsColumnCharWidth() CONST
virtual COLORREF GetTCHistoryFlagsColumnColor() CONST
virtual INT GetTCHistoryTCPacketColumnCharWidth() CONST
virtual COLORREF GetTCHistoryTCPacketColumnColor() CONST
virtual INT GetTCHistoryDispatchTimeColumnCharWidth() CONST
virtual COLORREF GetTCHistoryDispatchTimeColumnColor() CONST
virtual INT GetTCHistoryReleaseTimeColumnCharWidth() CONST
virtual COLORREF GetTCHistoryReleaseTimeColumnColor() CONST
virtual INT GetTCHistoryReleaseTimeOffsetColumnCharWidth() CONST
virtual COLORREF GetTCHistoryReleaseTimeOffsetColumnColor() CONST
virtual INT GetTCHistoryTransmissionTimeColumnCharWidth() CONST
virtual COLORREF GetTCHistoryTransmissionTimeColumnColor() CONST
virtual INT GetTCHistoryExecutionTimeColumnCharWidth() CONST
virtual COLORREF GetTCHistoryExcutionTimeColumnColor() CONST
virtual INT GetTCHistoryExecutionTimeOffsetColumnCharWidth() CONST
virtual COLORREF GetTCHistoryExcutionTimeOffsetColumnColor() CONST
virtual INT GetTCHistoryAuthorizationColumnCharWidth() CONST
virtual COLORREF GetTCHistoryAuthorizationColumnColor() CONST
virtual INT GetTCHistoryPTVTimeWindowColumnCharWidth() CONST
virtual COLORREF GetTCHistoryPTVTimeWindowColumnColor() CONST
virtual INT GetTCHistoryPTVChecksColumnCharWidth() CONST
virtual COLORREF GetTCHistoryPTVChecksColumnColor() CONST
virtual INT GetTCHistoryPTVResultColumnCharWidth() CONST
virtual COLORREF GetTCHistoryPTVResultColumnColor() CONST
virtual INT GetTCHistoryPEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCHistoryPEVChecksColumnColor() CONST
virtual INT GetTCHistoryPEVResultColumnCharWidth() CONST
virtual COLORREF GetTCHistoryPEVResultColumnColor() CONST
virtual INT GetTCHistoryCEVTimeWindowColumnCharWidth() CONST
virtual COLORREF GetTCHistoryCEVTimeWindowColumnColor() CONST
virtual INT GetTCHistoryCEVChecksColumnCharWidth() CONST
virtual COLORREF GetTCHistoryCEVChecksColumnColor() CONST
virtual INT GetTCHistoryCEVResultColumnCharWidth() CONST
virtual COLORREF GetTCHistoryCEVResultColumnColor() CONST
virtual INT GetTCHistoryStatusColumnCharWidth() CONST
virtual COLORREF GetTCHistoryStatusColumnColor() CONST
```

3.1.2.7. Database Table Management

Because the database driver is also responsible for the rendering all database content, there are a number of procdures to be implemented that handle the related tasks.

The following chapter lists all procedures that need to be implemented.

```
virtual BOOL CreateTables(CONST CDatabaseTableView
          *pView,ULONGLONG nComponent)
```

Creates all underlying (internal) data structures for the specified database component.

Parameters:

  pView

      Supplies a pointer to the view that contains the database tables.

  nComponent

      Specifies the database component for which the data structures should be created.

Return Value:

Returns TRUE if the required database table data structures were created successfully; FALSE if otherwise.

```
virtual BOOL SaveTables(CONST CDatabaseTableView *pView,ULONGLONG
          nComponent,BOOL bAll=FALSE,BOOL bPrompt=FALSE)
```

Saves the content of all tables related to the specified database component to the relational DBMS.

Parameters:

  pView

      Supplies a pointer to the view that contains the database tables.

  nComponent

      Specifies the database component for which all table data should be saved.

  bAll

      Specifies if only the current database table data item or all modified ones should be saved.

  bPrompt

      Specifies if the user should be asked for permission before saving.

Return Value:

Returns TRUE if the required database tables were saved successfully; FALSE if otherwise.

```
virtual BOOL CanSaveTables(CONST CDatabaseTableView
          *pView,ULONGLONG nComponent) CONST
```

Indicates if there is any data that has been modified for the specified database component.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

nComponent
> Specifies the database component for which all data should be checked for modification.

Return Value:

Returns TRUE if there is modified data for the specified database component that could be saved; FALSE if otherwise.

```
virtual BOOL CheckTables(CONST CDatabaseTableView *pView,ULONGLONG
          nComponent,CStringArray &szIssues) CONST
```

Checks the currently selected item of the specified database component for correctness and completeness.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

nComponent
> Specifies the database component for which all table data should be checked for correctness and completeness.

szIssues
> Contains a list of issues found in the data of the related database tables.

Return Value:

Returns TRUE if no issues (errors or warnings) were found for the currently selected item of the specified database component; FALSE if otherwise.

```
virtual BOOL CanCheckTables(CONST CDatabaseTableView
          *pView,ULONGLONG nComponent) CONST
```

Indicates if the currently selected item of the specified database component can be checked for correctness and completeness.

Parameters:
    pView
        Supplies a pointer to the view that contains the database tables.
    nComponent
        Specifies the database component those tables should be checked for correctness and completeness.

Return Value:
Returns TRUE if the tables of the specified database component could be checked for correctness and completeness; FALSE if otherwise.

```
virtual BOOL ImportTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent,BOOL bCustom=FALSE)
```

Imports data related to the specified database component from an external source; requires some user interaction.

Parameters:
    pView
        Supplies a pointer to the view that contains the database tables.
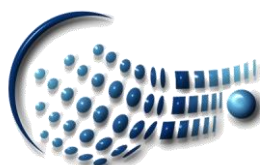   *nComponent*
        Specifies the database component for which data should be imported.
    bCustom
        Specifies if a non-standard procedure should be used for the data import.

Return Value:
Returns TRUE if data was imported successfully for the specified database component; FALSE if otherwise.

```
virtual BOOL CanImportTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent,BOOL bCustom=FALSE) CONST
```

Indicates if data can be imported for the specified database component.

Parameters:
    pView
        Supplies a pointer to the view that contains the database tables.
    nComponent
        Specifies the database component for which data should be imported.
    bCustom
        Specifies if a non-standard procedure should be used for the data import.

Return Value:

Returns TRUE if data could be imported for the specified database component; FALSE if otherwise.

```
virtual BOOL ExportTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent,BOOL bCustom=FALSE)
```

Exports data related to the specified database component to an external destination; requires some user interaction.

Parameters:

pView

> Supplies a pointer to the view that contains the database tables.

nComponent

> Specifies the database component for which data should be exported.

bCustom

> Specifies if a non-standard procedure should be used for the data export.

Return Value:

Returns TRUE if data was exported successfully for the specified database component; FALSE if otherwise.

```
virtual BOOL CanExportTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent,BOOL bCustom=FALSE) CONST
```

Indicates if data can be exported from the specified database component.

Parameters:

pView

> Supplies a pointer to the view that contains the database tables.

nComponent

> Specifies the database component for which data should be exported.

bCustom

> Specifies if a non-standard procedure should be used for the data export.

Return Value:

Returns TRUE if data could be exported from the specified database component; FALSE if otherwise.

```
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CDatabaseTMPackets &pDatabaseTMPackets)
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CDatabaseTCFunctions &pDatabaseTCFunctions)
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CDatabaseTCSequences &pDatabaseTCSequences)
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CDatabaseOBProcessors &pDatabaseOBProcessors)
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CDatabaseTMParameters &pDatabaseTMParameters)
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CDatabaseTCParameters &pDatabaseTCParameters)
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CANDLayouts &pANDLayouts)
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CGRDLayouts &pGRDLayouts)
virtual VOID UpdateTables(CONST CDatabaseTableView *pView,CONST
        CPODLayouts &pPODLayouts)
virtual BOOL UpdateTables(CDatabaseTableView *pView,ULONGLONG
        nComponent,LPCTSTR pszName,BOOL bDelete);[1]
virtual BOOL UpdateTables(CDatabaseTableView *pView,ULONGLONG
        nComponent,INT nID,BOOL bDelete);[1]
virtual VOID UpdateTables(CDatabaseTableView *pView,ULONGLONG
        nComponent);[2]
virtual VOID UpdateTables(CDatabaseTableView *pView);[3]
```

Updates all data related to the specified database component with the supplied data.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

pDatabaseTMPackets
> Specifies the telemetry packets to be used for the update.

pDatabaseTCFunctions
> Specifies the telecommand functions to be used for the update.

pDatabaseTCSequences
> Specifies the telecommand sequences to be used for the update.

pDatabaseOBProcessors
> Specifies the on-board processors to be used for the update.

pDatabaseTMParameters
> Specifies the telemetry parameters to be used for the update.

pDatabaseTCParameters
> Specifies the telecommand parameters to be used for the update.

pANDLayouts
> Specifies the alphanumeric displays to be used for the update.

pGRDLayouts
> Specifies the graphic displays to be used for the update.

pPODLayouts
> Specifies the parameter observation displays to be used for the update.

nComponent
> Specifies the database component for which data should be updated.

pszName
> Specifies the name of the database component item that should be updated.

nID
> Specifies the identification number of the database component item that should be updated.

bDelete
> Specifies that the database component item should be deleted.

📝 Note:

[1]The specified database component item is updated by the data contained in the `CDatabaseEngine`-derived class (unless *bDelete* is TRUE).

[2]All table fields related to the specified database component are enumerated with current (updated) values.

[3]Performs special table updates.

Return Value:

None except for [1] where TRUE is returned if the specified database component item was updated (or deleted) successfully; FALSE if not.

```
virtual VOID ArrangeTables(CONST CDatabaseTableView *pView)
virtual VOID ArrangeTables(CONST CDatabaseTableView *pView,UINT
        nControlID)
```

Arranges the tables inside the view.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.
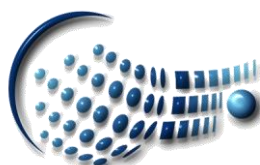
nControlID
> Specifies the identification number of the control causing the arrangement of the tables.

Return Value:
None.

```
virtual BOOL DiscardTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent,BOOL bAll=FALSE)
```

Discards (empties) all tables related to the specified database component without deleting any data.

Parameters:
pView
> Supplies a pointer to the view that contains the database tables.

nComponent
> Specifies the database component for which the content of the tables should be discarded.

bAll
> Specifies if only the top-level table's content should be discarded.

Return Value:
Returns TRUE if the content of the specified database component's tables was discarded successfully; FALSE if otherwise.

```
virtual BOOL CanDiscardTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent,BOOL bAll=FALSE) CONST
```

Indicates if all tables related to the specified database component can be discarded.

Parameters:
pView
> Supplies a pointer to the view that contains the database tables.

nComponent
> Specifies the database component for which the content of the tables should be discarded.

bAll
> Specifies if only the top-level table's content should be discarded.

Return Value:
Returns TRUE if the content of the specified database component's tables could be discarded; FALSE if otherwise.

```
virtual BOOL DeleteTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent)
```

Deletes the currently selected database component item and empties the related tables.

Parameters:

pView

    Supplies a pointer to the view that contains the database tables.

nComponent

    Specifies the database component for which the currently selected item should be deleted.

Return Value:

Returns TRUE if the specified database component's currently selected item was deleted successfully; FALSE if otherwise.

```
virtual BOOL CanDeleteTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent) CONST
```

Indicates if the currently selected database component item can be deleted.

Parameters:

pView

    Supplies a pointer to the view that contains the database tables.

nComponent

    Specifies the database component for which the currently selected item should be deleted.

Return Value:

Returns TRUE if the specified database component's currently selected item could be deleted; FALSE if otherwise.

```
virtual BOOL DestroyTables(CONST CDatabaseTableView
        *pView,ULONGLONG nComponent)
```

Destroys all underlying (internal) data structures for the specified database component.

Parameters:

pView

    Supplies a pointer to the view that contains the database tables.

nComponent

    Specifies the database component for which the data structures should be destroyed.

Return Value:

Returns TRUE if the required database table data structures were destroyed successfully; FALSE if otherwise.

```
virtual INT EnumTables(CONST CDatabaseTableView
        *pView,CStringArray &szTables,CUIntArray &nTypes,BOOL
        bMenu,BOOL bResetable=FALSE) CONST
```

Enumerates all database tables.

Parameters:

pView

Supplies a pointer to the view that contains the database tables.

szTables

Contains the names of the enumerated tables.

nTypes

Contains the type of the enumerated tables:

Type	Description
TAS_TABLE_NORMAL	Specifies a top-level table with a single value column that can be the root of related tables.
TAS_TABLE_LIST	Specifies a table with a fixed column layout containing one or more entries.

Return Value:

Returns the number of tables enumerated; -1 if an error occurred.

```
virtual BOOL GetTableTitle(CONST CDatabaseTableView *pView,LPCTSTR
        pszTable,CString &szTitle) CONST
```

Returns the title of a database table.

Parameters:

pView

Supplies a pointer to the view that contains the database tables.

pszTable

Specifies the table to get the title for.

szTitle

Contains the title of the table.

Return Value:

Returns TRUE if a title is returned; FALSE if not.
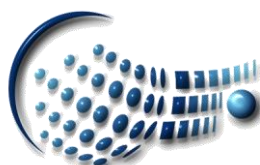
```
virtual BOOL GetTableLayout(CONST CDatabaseTableView
          *pView,LPCTSTR pszTable,CUIntArray &nWidths) CONST
virtual BOOL GetTableLayout(CONST CDatabaseTableView
          *pView,LPCTSTR pszTable,LPCTSTR pszItem,CUIntArray
          &nWidths) CONST
```

Returns the column widths for the specified table.

Parameters:

pView

Supplies a pointer to the view that contains the database tables.

pszTable

Specifies the table for which to get the column layout.

pszItem

Specifies the database component item currently displayed in the table.

Note:

Some tables need to adjust their column widths depending on the content displayed.

nWidths

Contains the column widths (in pixels) of the table.

Return Value:

Returns TRUE if the column widths are returned; FALSE if not.

```
virtual BOOL GetTablePosition(CONST CDatabaseTableView
          *pView,LPCTSTR pszTable,CString &szTable,CRect
          &rPosition) CONST
```

Returns the position of the specified database table inside the view.

Parameters:

pView

Supplies a pointer to the view that contains the database tables.

pszTable

Specifies the table to get the position for.

szTable

Contains the parental table to which the table is linked to. If there is no such table, the name specified by *pszTable* is returned.

rPosition
    Contains the position of the table:

rPosition.left = *n* (> 0)	The table is positioned to the left of the table returned in *szTable* and vertically aligned with row *n*.
rPosition.top = *n* (> 0)	The table is located above its parental table, horizontally aligned with column *n*.
rPosition.right = *n* (> 0)	The table is positioned to the right of the table returned in *szTable* and vertically aligned with row *n*.
rPosition.bottom = *n* (> 0)	The table is located below its parental table, horizontally aligned with column *n*.

Return Value:
Returns TRUE if a position is returned for the table; FALSE if not.

```
virtual BOOL GetTableOutline(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,COLORREF
        &nTitleRowColor,COLORREF &nFirstRowColor,COLORREF
        &nFirstColumnColor,COLORREF &nTitleFontColor,COLORREF
        &nItemsFontColor,INT &nOuterBorderSize,INT
        &nOuterBorderStyle,COLORREF &nOuterBorderColor,INT
        &nInnerBorderSize,INT &nInnerBorderStyle,COLORREF
        &nInnerBorderColor) CONST
```

Returns the outline of the specified database table inside the view.
Parameters:
pView
    Supplies a pointer to the view that contains the database tables.
pszTable
    Specifies the table to get the outline for.
nTitleRowColor
    Contains the color used to paint the background of the title row of the table.
nFirstRowColor
    Contains the color used to paint the background of the first row of the table.
nFirstColumnColor
    Contains the color used to paint the background of the first column of the table.

nTitleFontColor
     Contains the color used to draw the text of the table title row.
nItemsFontColor
     Contains the color used to draw the text of the non-title table rows.
nOuterBorderSize
     Contains the thickness (in pixels) of the outer table border.
nOuterBorderStyle
     Contains the style of the outer table border.
nOuterBorderColor
     Contains the color of the outer table border.
nInnerBorderSize
     Contains the thickness (in pixels) of the inner table border.
nInnerBorderStyle
     Contains the style of the inner table border.
nInnerBorderColor
     Contains the color of the inner table border.

Return Value:
Returns TRUE if an outline is returned for the specified table; FALSE if not.

```
virtual BOOL GetTableOutlineFonts(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LOGFONT *plfTitleFont,LOGFONT
        *plfItemsFont) CONST
```

Returns the fonts of the specified database table inside the view.
Parameters:
pView
     Supplies a pointer to the view that contains the database tables.
pszTable
     Specifies the table to get the fonts for.
plfTitleFont
     Contains the font to be used for the table title row.
plfItemsFont
     Contains the font to be used for the non-title table rows.

Return Value:
Returns TRUE if the fonts are returned for the specified table; FALSE if not.

```
virtual BOOL GetTableOutlineLimits(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,INT &nMinRows,INT &nMaxRows)
        CONST
```

Returns the fonts of the specified database table inside the view.

Parameters:

pView
>   Supplies a pointer to the view that contains the database tables.

pszTable
>   Specifies the table to get the limits for.

nMinRows
>   Contains the minimum number of rows that the table must contain.

nMaxRows
>   Contains the maximum number of rows that the table can contain.

Return Value:

Returns TRUE if the outline limits are returned for the specified table; FALSE if not.

```
virtual BOOL GetTableContents(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszItem,INT
        nRows,CStringArray &szContents,CStringArray
        &szDescriptions,CStringArray &szDataTypes,CUIntArray
        &nFormats,CUIntArray &nIDs,BOOL &bResize,BOOL &bVisible)
        CONST
```

Returns the contents of the specified database table inside the view.

Parameters:

pView
>   Supplies a pointer to the view that contains the database tables.

pszTable
>   Specifies the table to get the contents for.

pszItem
>   Specifies the database content to be displayed in the table.

szContents
>   Contains the table row contents of the table.

>   Note:
>   Each row is returned as a TAB-separated entry in the string array to indicate the columns.

szDescriptions

Contains a description of the table row or column depending on whether the table if of type TAS_TABLE_NORMAL or TAS_TABLE_LIST.

szDatatypes

Contains a description of the alterable value datatype of the row or column depending on the table type TAS_TABLE_NORMAL or TAS_TABLE_LIST.

nFormats

Contains the format of the alterable value datatype:

Type	Description
TAS_TYPE_TEXT	Specifies a simple non-modifiable text.
TAS_TYPE_EDIT	Specifies a single-line text edit field.
TAS_TYPE_DPLEDIT	Specifies an editable area that accepts code written in the **Derived Parameter Language** (DPL).
TAS_TYPE_MDLEDIT	Specifies an editable area that accepts code written in the **Mimics Description Language** (MDL).
TAS_TYPE_TPLEDIT	Specifies an editable area that accepts code written in the **Telecommand Procedure Language** (TPL).
TAS_TYPE_MULTILINEEDIT	Specifies a multiline-line text edit field.
TAS_TYPE_DECNUMBEREDIT	Specifies a numerical edit field using a decimal radix.
TAS_TYPE_RAWNUMBEREDIT	Specifies a numerical edit field with a definable radix (2, 8, 10 or 16).
TAS_TYPE_COMBOLISTBOX	Specifies a combo box (list only).
TAS_TYPE_COMBOEDITBOX	Specifies a combo box (list and edit capabilities).
TAS_TYPE_COMBOCOLORSBOX	Specifes a combo box displaying chooseable colors.
TAS_TYPE_COMBOLINESTYLESBOX	Specifies a combo box displaying chooseable line styles.
TAS_TYPE_SPINBOX	Specifies a spin box for numbers.
TAS_TYPE_TIMESPINBOX	Specifies a time spin box.
TAS_TYPE_SUBTABLE	Specifies a sub-table.

nIDs
>   Contains a unique identifier for the control of the alterable value.

bResize
>   Indicates if the table layout can be resized to adjust the content.

bVisible
>   Indicates if the table is visible.

Return Value:

Returns TRUE if the contents is returned for the specified table; FALSE if not.

```
virtual INT EnumTableFieldValues(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszField,CStringArray
        &szValues,CLongUIntArray &nValues) CONST
```

Parameters:

pView
>   Supplies a pointer to the view that contains the database tables.

pszTable
>   Specifies the table to enumerate available values for a certain changeable edit field.

pszField
>   Specifies the alterable table field those valid values should be enumerated.

szValues
>   Contains the values enumerated in a textual format.

nValues
>   Contains the values enumerated in a numerical format.

Return Value:

Returns the number of enumerated values for the specified table field; -1 if an error occurred.

```
virtual BOOL GetTableFieldLimits(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszField,INT &nLow,INT
        &nHigh) CONST
virtual BOOL GetTableFieldLimits(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszField,UINT &nLow,UINT
        &nHigh) CONST
virtual BOOL GetTableFieldLimits(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszField,double
        &fLow,double &fHigh) CONST
```

Parameters:
  pView
        Supplies a pointer to the view that contains the database tables.
  pszTable
        Specifies the table to enumerate available values for a certain changeable edit field.
  pszField
        Specifies the table field for which to get the limits.
  nLow
  fLow
        Contains the lower limit of the numerical edit field.
  nHigh
  fHigh
        Contains the higher limit of the numerical edit field.

Return Value:
Returns TRUE if the table field limits are returned; FALSE if not.

```
virtual BOOL GetTableContext(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable, CString &szTable,UINT
        &nControlID,CString &szValue) CONST
```

Returns the current context of the specified database table inside the view.
Parameters:
  pView
        Supplies a pointer to the view that contains the database tables.
  pszTable
        Specifies the table to get the current context for.
  szTable
        Contains the parental table to which the table is linked to. If there is no such table,
        the name specified by *pszTable* is returned.

nControlID
> Contains the identification number of the control definng the content of the table.

szValue
> Contains the name of the database component displayed in the table.

Return Value:

Returns TRUE if a context is returned for the table; FALSE if not.

```
virtual INT EnumTableUpdates(CONST CDatabaseTableView *pView,UINT
        nControlID,CStringArray &szTables) CONST
```

Enumerates the tables those content needs to be updated when the specified table field is changed.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

nControlID
> Specifies the identification number of the control attached to the table field that changed.

szTables
> Contains the tables that need to be updated.

Return Value:

Returns the number of enumerated tables; -1 if an error occurred.

```
virtual BOOL GetTableUpdateContents(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,UINT nControlID,LPCTSTR
        pszValue,CString &szItem,CStringArray &szContents) CONST
```

Returns the current context of the specified database table inside the view depending on the value of ab editable control.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

pszTable
> Specifies the table to get the updated content for.

nControlID
> Specifies the identification number of the control that caused the table update.

pszValue
> Specifies the current (new) value for the control.

szItem
>   Contains the name of database component (context) returned.

szContents
>   Contains the contents of the database component.

Return Value:

Returns TRUE if the contents is returned for the table; FALSE if not.

```
virtual BOOL UpdateTableFieldData(CONST CDatabaseTableView *pView,
        UINT nControlID,LPCTSTR pszValue,LPCTSTR
        pszOldValue=NULL)
```

Updates the internal data representation of a table.

Parameters:

pView
>   Supplies a pointer to the view that contains the database tables.

nControlID
>   Specifies the identification number of the control that changed.

pszValue
>   Specifies the current (new) value for the control.

pszOldValue
>   Specifies the previous (old) value for the control.

Return Value:

Returns TRUE if the table data could be updated; FALSE if not.

```
virtual INT InsertTableFieldData(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszValue)
virtual BOOL InsertTableFieldData(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszValue,INT nRow,INT
        nCol=-1)
```

Notifies that data is inserted into a table or inserts a row or column.

Parameters:

pView
>   Supplies a pointer to the view that contains the database tables.

pszTable
>   Specifies the table to receive data, a new row or column.

pszValue
>   Specifies the name of the database component for which data is inserted.

nRow
> Specifies the position where to insert the new row.

nCol
> Specifies the position where to insert the new column.

Return Value:

Returns the index of the database component; -1 if an error occurred. For the latter two procedures TRUE is returned when the row or column could be inserted, FALSE if not.

```
virtual BOOL CheckTableFieldData(CONST CDatabaseTableView *pView,
        LPCTSTR pszTable,BOOL bModified=FALSE,BOOL bExist=FALSE)
        CONST
virtual BOOL CheckTableFieldData(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,CStringArray &szIssues) CONST
```

Checks the content of the table for correctness, modifications or existence.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

pszTable
> Specifies the table those content should be checked.

bModified
> Checks for a modification of the table contents (if TRUE).

bExist
> Checks for existance of the current table contents (if TRUE).

szIssues
> Contains all issues (as human readable text) found when checking the specified table.

Return Value:

Returns TRUE if the table contents check was successful; FALSE if not.

```
virtual BOOL SelectTableFieldData(CONST CDatabaseTableView *pView,
        LPCTSTR pszTable,CONST POINT &pt)
virtual BOOL SelectTableFieldData(CONST CDatabaseTableView *pView,
        LPCTSTR pszTable,LPCTSTR pszValue)
```

Selects a row or column in a table and performs the appropriate rendering.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

pszTable
> Specifies the table in which a row or column should be selected.

pt
> Specifies position (in pixels) inside the table to be selected.

pszValue
> Specifies the database component selected for the table.

Return Value:

Returns TRUE if the table selection succeeded; FALSE if not.

```
virtual BOOL SaveTableFieldData(CONST CDatabaseTableView *pView,
        LPCTSTR pszTable,BOOL bPrompt=FALSE)
```

Saves the contents of a table and optionally asks for confirmation.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

pszTable
> Specifies the table to be saved.

bPrompt
> Pops-up with a dialog box asking for confirmation (if TRUE).

Return Value:

Returns TRUE if the table could be saved successfully; FALSE if not.

```
virtual BOOL ProtectTableFieldData(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable ,BOOL bUndo=FALSE)
```

Changes the inteactivity of a table by enabling or disabling any input on edit fields depending on the current contents or changes the rendering.

Parameters:

pView
> Supplies a pointer to the view that contains the database table.

pszTable
> Specifies the table to be protected.

bUndo
> Specifies if the action performed by the function should be undoable or not.

Return Value:

Returns TRUE if the table could be protected successfully; FALSE if not.

```
virtual BOOL DeleteTableFieldData(CONST CDatabaseTableView *pView,
        LPCTSTR pszTable)
virtual INT DeleteTableFieldData(CONST CDatabaseTableView *pView,
        LPCTSTR pszTable,LPCTSTR pszValue)
virtual BOOL DeleteTableFieldData(CONST CDatabaseTableView *pView,
        LPCTSTR pszTable,LPCTSTR pszValue,INT nRow,INT nCol=-1)
```

Notifies that data is deleted from a table or deletes a row or column.

Parameters:

   pView
      Supplies a pointer to the view that contains the database tables.
   pszTable
      Specifies the table to get data, a row or column deleted.
   pszValue
      Specifies the name of the database component for which data is deleted.
   nRow
      Specifies the position where to delete the row.
   nCol
      Specifies the position where to delete the column.

Return Value:

Returns the index of the database component deleted; -1 if an error occurred. For the latter two procedures TRUE is returned when the row or column could be deleted, FALSE if not.

```
virtual BOOL GetTableFieldFlag(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,INT nRow,INT nCol,CString
        &szFlag) CONST
```

Returns a textual flag indicating if a table field is mandatory or not.

Parameters:

   pView
      Supplies a pointer to the view that contains the database tables.
   pszTable
      Specifies the table containing the field for which to get the information.

nRow
> Specifies the row position of the field.

nCol
> Specifies the column position of the field.

szFlag
> Contains the flag for the specified field.

Return Value:

Returns TRUE if the flag is returned for the specified field; FALSE if not.

```
virtual INT EnumTableDropTargets(CONST CDatabaseTableView *pView,
        ULONGLONG nComponent,CStringArray &szTables,CStringArray
        &szFields) CONST
```

Enumerates all tables containing drop targets for a certain database component.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

nComponent
> Specifies the database component for which to find the drop targets.
>
> See the 'BOOL **Save(…)**' procedure for all possible argument values.

szTables
> Contains the list of tables that have a drop target of the specified type.
>
> Note:
>
> A table may be listed multiple times i.e. once for each field.

szFields
> Contains the list of table fields that can act as a drop target.

Return Value:

Returns the number of enumerated drop targets; -1 if an error occurred.

```
virtual BOOL FindTableDropTargetCell(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszField,ULONGLONG
        nComponent,LPCTSTR pszItem,CString &szItem,INT &nRow,INT
        &nCol) CONST
```

Returns the identification and position of the field that can act as a drop target.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.
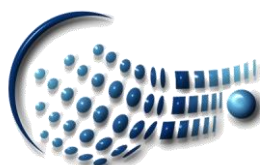
pszTable
>> Specifies the table containing the field that can act as a drop target.

pszField
>> Specifies the field.

nComponent
>> Specifies the type of the drop target.
>> See the 'BOOL **Save(…)**' procedure for all possible argument values.

pszItem
>> Specifies the database component acting as drop object.

szItem
>> Contains a displayable string identifying the database component being dropped.

nRow
>> Contains the row position of the field where the dropping will take place.

nCol
>> Specifies the column position of the field.

Return Value:

Returns TRUE if a suitable drop target field is returned; FALSE if not.

```
virtual INT EnumPrintTables(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,UINT nFlags,CStringArray
        &szTables) CONST
```

Enumerates all associated tables that need to be printed together with a specified table.

Parameters:

pView
>> Supplies a pointer to the view that contains the database tables.

pszTable
>> Specifies the table going to be printed.

nFlags
>> Specifies how the tables should to be printed:

Flag	Description
DATABASETABLEVIEW_PRINT_TABLE	Enumerates tables of type TAS_TYPE_NORMAL only.
DATABASETABLEVIEW_PRINT_SUBTABLE	Enumerates tables that are not of the above mentioned type.

DATABASETABLEVIEW_PRINT_ALL	All available database components related to the specified table are enumerated.

szTables
>   Contains a list of all tables that need to be printed together with the specified one.

Return Value:
Returns the number of tables enumertated; -1 if an error occurred.

```
virtual BOOL GetPrintTableContents(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszItem,CStringArray
        &szContents) CONST
```

Gets the content to be printed for a specified table.

Parameters:
pView
>   Supplies a pointer to the view that contains the database tables.

pszTable
>   Specifies the table going to be printed.

pszItem
>   Specifies the database component to be printed.

szContents
>   Contains the contents of the database component.

Return Value:
Returns TRUE if the contents to be printed is returned; FALSE if not.

```
virtual BOOL PrintTable(CONST CDatabaseTableView *pView,LPCTSTR
        pszTable,LPCTSTR pszItem,CMetaFileDC &cMetaDC)
```

Prints the table with the specified database component as content.

Parameters:
pView
>   Supplies a pointer to the view that contains the database tables.

pszTable
>   Specifies the table (incl. associated ones) to be printed.

pszItem
>   Specifies the database component (content).

cMetaDC
> Contains a device context with the printable table.

Return Value:
Returns TRUE if the table was printed successfully; FALSE if not.

```
virtual ULONGLONG TranslateTableNameToComponent(CONST
        CDatabaseTableView *pView,LPCTSTR pszTable) CONST
```

Associates a table with a database component.
Parameters:
pView
> Supplies a pointer to the view that contains the database tables.

pszTable
> Specifies the table to associate with a database component.

Return Value:
Returns the database component associated with the specified table; 0 if an error occurred.

```
virtual CString TranslateComponentToTableName(CONST
        CDatabaseTableView *pView,ULONGLONG nComponent) CONST
```

Associates a database component with a table.
Parameters:
pView
> Supplies a pointer to the view that contains the database tables.
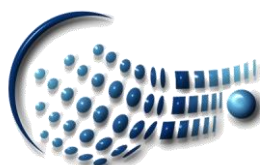
nComponent
> Specifies the database component to associate with a table.

Return Value:
Returns the table associated with the specified database component; an empty string if an error occurred.

```
virtual INT TranslateTableFieldToIndex(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszField) CONST
```

Finds the table-based index for a specified field.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

pszTable
> Specifies the table that contains the field to get the index for.

pszField
> Specifies the field.

Return Value:

Returns the index for a specified field; -1 if an error occurred.

```
virtual INT TranslateTableFieldToID(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,LPCTSTR pszField) CONST
```

Finds the control identification number for a specified field.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

pszTable
> Specifies the table that contains the field to get the identification number for.

pszField
> Specifies the field.

Return Value:

Returns the control identification number for a specified field; -1 if an error occurred.

```
virtual BOOL IsTableFieldID(CONST CDatabaseTableView
        *pView,LPCTSTR pszTable,UNIT nControlID) CONST
```

Checks if an identification number belongs to a control hosted by a specified table.

Parameters:

pView
> Supplies a pointer to the view that contains the database tables.

pszTable
> Specifies the table that is checked for the control identification number.

nControlID
> Specifies the control identification number.

Return Value:

Returns TRUE if the identification number belongs to a control; FALSE if otherwise.

3.1.2.8. DBMS Tables Exposure

The following functions are related to the *Database Management System* (DBMS) i.e. they are used for the interaction with the DBMS:

```
virtual INT EnumDBMSTables(CStringArray &szTables,CStringArray
        &szDescriptions) CONST
```

Enumerates all database tables (including a short description) that are part of the relational DBMS-based database.

Parameters:

szTables
>Contains the names of all tables returned.

szDescriptions
>Contains a description for the corresponding tables.

Return Value:

Returns the number of tables enumerated.

```
virtual BOOL GetDBMSTableCharacteristics(LPCTSTR
        pszTable,CStringArray &szColumns,CStringArray
        &szDataTypes,CByteArray &bAllowNulls) CONST
```

Returns the characteristics for a specified relational DBMS table.

Parameters:

pszTable
>Specifies the table those characteristics should be returned.

szColumns
>Contains the columns the specified table consists of.

szDataTypes
>Contains the data type for the corresponding columns.
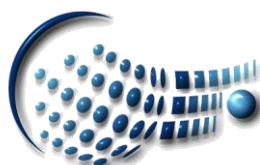
bAllowNulls
>Contains a flag indicating if a null value is allowed for the corresponding columns.

Return Value:

Returns TRUE if the characteristics were returned for the specified table; FALSE if not.

```
virtual BOOL UsesDBMSTables(ULONGLONG nComponent) CONST
```

Indicates if the specified database component is using relational DBMS-based tables.

Parameters:

nComponent

Specifies the database component to be checked for its relational DBMS table usage.

Return Value:

Returns TRUE if the specified database component uses relational DBMS tables; FALSE if not.

3.1.3. Parsing Database Functions

Additional functions are provided to help parsing number & time strings:

```
static BOOL ParseValue(LPCTSTR pszValue,ULONGLONG &nValue);
static BOOL ParseValue(LPCTSTR pszValue,UINT nRadix,ULONGLONG
        &nValue);
static BOOL ParseValue(LPCTSTR pszValue,UINT nRadix,ULONGLONG
        &nValue,BOOL &bSign);
static ULONGLONG ParseValue(LPCTSTR pszValue,UINT nRadix);
static BOOL ParseValue(LPCTSTR pszValue,double &fValue);
static double ParseValue(LPCTSTR pszValue);
static TIMETAG ParseTime(LPCTSTR pszTime,BOOL bAbsolute=TRUE);
static TIMETAG ParseTime(CONST CByteArray &nCode,INT nPTC,INT
        nPFC);
static BOOL ParseTime(TIMETAG tTime,INT nPTC,INT nPFC,CByteArray
        &nCode);
```

Parses a number or time and returns its numerical value.

Parameters:

pszValue
pszTime

Specifies the number or time to be parsed.

nCode

Specifies the CDS or CUC encoded time (as specified by *nPTC* & *nPFC*).

tTime

Specifies the time as a number of microseconds since (January 1, 1970).

nPTC

Specifies the parameter type code (see chapter below).

nPFC

Specifies the parameter format code (see chapter below).

nRadix
>Specifies the radix of the number supplied.

nValue
fValue
>Contains the parsed (floating-point) value.

bSign
>Contains the sign flag (if negative).

bAbsolute
>Specifies if the supplied time string is in an absolute format.

Return Value:
Returns the number or time in a numerical format.

3.1.4. Encoding & Decoding Database Functions
Additional functions are provided to help encoding number & time values:

```
static ULONGLONG EncodeValue(UINT nType,ULONGLONG nAttributes,INT
        nWidth,ULONGLONG nValue,BOOL bTC=TRUE)
static ULONGLONG EncodeValue(UINT nType,ULONGLONG
        nAttributes,double fValue,BOOL bTC=TRUE)
static ULONGLONG EncodeTime(TIMETAG tTime)
```

Encodes numerical or time values.

Parameters:
nType
>Specifies the encoding format to be used for floating-point numbers:

Format	Description
DATABASE_TYPE_FLOATINGPOINTFORMAT_IEEE	Follows the IEEE 754 floating-point standard.
DATABASE_TYPE_FLOATINGPOINTFORMAT_USAF	Follows the MIL-STD-1750A standard of the US Air Force.

nAttributes
>Specifies the nature (applies to telemetry parameters only) of the value and the coding to be assumed:

Attribute	Description
TMPARAMETER_NATURE_RAW	The value is a normal telemetry parameter value.
TMPARAMETER_NATURE_CONSTANT	The value is a constant.
TMPARAMETER_NATURE_DERIVED	The value originates from a derived telemetry parameter.
TMPARAMETER_CODING_BITPATTERN TMPARAMETER_CODING_INVERSEBITPATTERN TMPARAMETER_CODING_SIGNEDINTEGER TMPARAMETER_CODING_UNSIGNEDINTEGER TMPARAMETER_CODING_COMPLEMENTCOMPLEX TMPARAMETER_CODING_FLOATINGPOINT TCPARAMETER_CODING_BITPATTERN TCPARAMETER_CODING_SIGNEDINTEGER TCPARAMETER_CODING_UNSIGNEDINTEGER TCPARAMETER_CODING_FLOATINGPOINT TCPARAMETER_CODING_TIME	The value should be interpreted as having the indicated coding.

nWidth
> Specifies the width (in bits) available for encoding.

nValue
> Specifies the value to be encoded.

tTime
> Specifies the time (in microseconds since January 1, 1970) to be encoded.

bTC
> Specifies if the value belongs to a telecommand parameter (if TRUE).

Return Value:

Returns the encoded value or time as specified.

```
static ULONGLONG DecodeValue(UINT nType,ULONGLONG nAttributes,INT
        nWidth,ULONGLONG nCode,BOOL bTC=TRUE)
static double DecodeValue(UINT nType,ULONGLONG
        nAttributes,ULONGLONG nCode,BOOL bTC=TRUE)
static TIMETAG DecodeTime(ULONGLONG nCode)
```

Decodes numerical or time values.

Parameters:

nType

Specifies the decoding format to be used for floating-point numbers.

See the 'ULONGLONG **EncodeValue(**…**)**' function for all possible argument values.

nAttributes

Specifies the nature (applies to telemetry parameters only) of the value and the coding to be assumed.

See the 'ULONGLONG **EncodeValue(**…**)**' function for all possible argument values.

nWidth

Specifies the width (in bits) available to decode.

nCode

Specifies the value to be decoded.

bTC

Specifies if the value belongs to a telecommand parameter (if TRUE).

Return Value:

Returns the decoded value as specified.

3.1.5. Type Code Conversion Database Functions

Additional functions are provided to help converting type code values:

```
static BOOL TranslateParameterTypeCode(INT nPTC,INT nPFC,BOOL
        bTC,ULONGLONG &nAttributes,INT &nWidth)¹
static CString TranslateParameterTypeCode(INT nPTC,INT nPFC,BOOL
        bTC)²
static CString TranslateParameterTypeCode(ULONGLONG
        nAttributes,BOOL bTC)²
```

Translates parameter type & format codes (SCOS-2000 specific) into internal codes.

Parameters:

nPTC
> Specifies the *Parameter Type Code* (PTC).
> See the SCOS-2000 Database Import ICD (EGOS-MCS-S2K-ICD-0001) for possible values.

nPFC
> Specifies the *Parameter Format Code* (PFC).
> See the SCOS-2000 Database Import ICD (EGOS-MCS-S2K-ICD-0001) for possible values.

bTC
> Specifies if the supplied PTC & PFC codes refer to a telemetry or telecommand parameter.

nAttributes
> Contains the converted code.

nWidth
> Contains the width (in bits) of the telemetry or telecommand parameter.

Return Value:
[1]Returns TRUE if there is a valid type code conversion; FALSE if not.
[2]Returns a human-readable text equivalent to the supplied parameter type & format (or internal) code.

3.1.6. Helper Database Functions
Some more functions provide additional support.

```
BOOL ShowMessage(CONST EVENT_DESCRIPTOR nMessageID)
BOOL ShowMessage(LPCTSTR pszMessage, BOOL bAudition=TRUE)
```

Shows a pre-defined message or a supplied message text.
Parameters:

nMessageID
> Specifies event descriptior number of the predefined message to be shown.

pszMessage
> Specifies the message text to be shown.

bAudition
> Specifies if an alertable sound should be generated when displaying the message.

Return Value:
Returns TRUE if the message could be displayed; FALSE if not.

```
BOOL ShowHelp(LPCTSTR pszTopic)
```

Opens the online help at the specified topic.

Parameters:

    pszTopic

        Specifies the topic in the online help to be opened.

Return Value:

Returns TRUE if the online help could be opened successfully; FALSE if not.

```
CString Required() CONST
CString Optional() CONST
CString NotApplicable() CONST
```

Retrieves the text to be displayed for a relational DBMS table field in order to indicate if it is required, optional or not applicable.

Return Value:

Returns the related text.

```
CString Char(INT nChars) CONST
CString Varchar(INT nChars) CONST
CString Text() CONST
CString Integer() CONST
CString BigInt() CONST
CString SmallInt() CONST
CString TinyInt() CONST
CString Float() CONST
CString Real() CONST
CString Double() CONST
CString Numeric(INT nDigits,INT nPrecision) CONST
CString Decimal(INT nDigits,INT nPrecision) CONST
CString Binary(INT nDigits) CONST
CString Varbinary(INT nDigits) CONST
CString Bit() CONST
CString Date() CONST
CString Time() CONST
CString Timestamp() CONST
```

Retrieves the text to be displayed for a relational DBMS table field in order to indicate its data type.

Parameters:

nChars
nDigits

Specifies the number of characters or digits.

nPrecision

Specifies the precision of a number.

Return Value:

Returns the data type as a text.

CDatabaseEngine ***GetDatabaseEngine()** CONST

Returns a pointer to the database currently open.

Return Value:

Returns the pointer to the **CDatabaseEngine**-derived class.

CDatabase ***GetDBMS()** CONST

Returns a pointer to the DBMS hosting the database currently open.

Return Value:

Returns the pointer to the **CDatabase**-derived class.

4. Software Prerequisites

The Microsoft® Visual Studio® 2010 or higher is required as well as the *SatView™ Database Driver Development Kit* (DDK) which is available for free for all customers of **BINARY SPACE**.

5. Samples

The SatView™ Database DDK contains a complete implementation of a database driver for the ALTEL database standard used to support ARTEMIS.