

SMS Broker System
Software Integration

Our system uses
WCF
(Windows Communication Foundation)
&
.NET

**Statement
classes and interfaces.**

(ver. from 8/3/2026)

Table of Contents

1. Special Notes.	1
2. Daily Statement.	1
2.1 Daily Statement Entry.	6
2.1.1 Daily Statement Entry Fee.	8
2.2 Daily Statement Fee.	8
3. Monthly Statement.	9
3.1 Monthly Statement Entry.	11
3.2 Statement Monthly Fee.	12
3.3 Daily Statement Entry Fee.	12

1. Special Notes.

1. In contrast to other classes ([CustomsEntry](#), [Shipment](#), [ImporterSecurityFilling](#), [Drawback](#), [In-Bond](#), etc.) daily and monthly statements classes ([StatementDaily](#) and [StatementMonthly](#)) objects are added to SMS-database by ABI part of SMS-system when PF-App and MS-app files come from Customs. Hence, statement manager interfaces doesn't suggest New(), Save() and ValidateTreeByld() methods available.
2. [StatementDaily](#) class is used for periodic and non-periodic daily statements.

2. Daily Statement.

Statement document contains an info about duties and fees owed to CBP for the entries.

Class: [DataContract](#)

SMS.Broker.DataContracts.Documents.[StatementDaily](#) : [EntityDocument](#)
Header data of Statement document.

Properties:

Attributes	Name	Type	Description
[DataMember] [Column(TypeName = "date")]	AcceptedDate	DateTime?	The date of successful statement authorization.
[DataMember] [DecimalPrecisionAttribute(11,	AmountDue	decimal	Total Amount Due.

2)]			
[DataMember] [StringLength(20)]	ClientRef	string	Client reference.
[DataMember] [InverseProperty("StatementDaily")]	CustomsEntries	List<StatementDailyEntry>	The daily statement entry (entry info) collection.
[DataMember] [StringLength(3)]	EntryFilerCode	string	Entry filer code.
[DataMember] [InverseProperty("StatementDaily")]	FeeAmounts	List<StatementDailyFee>	The collection of fees.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalAmountDue	decimal	Final Total Amount Due.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalAntidumpingDuty	decimal	Final total antidumping duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalCountervailingDuty	decimal	Final total countervailing duty amount.
[DataMember] [Column(TypeName = "date")]	FinalDate	DateTime?	The date when final daily statement is issued by Customs.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalDeferredTax	decimal	Final total deferred tax amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalEstimatedDuty	decimal	Final total estimated duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalEstimatedTax	decimal	Final total estimated tax amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalInterestAmountReconciliationSummary	decimal	Final total interest amount for reconciliation summary.
[DataMember] [DecimalPrecisionAttribute(5, 0)]	FinalNumberNonRevenueProducingEntries	decimal	Final total number of non-revenue producing entries in the transaction.
[DataMember] [DecimalPrecisionAttribute(5, 0)]	FinalNumberRevenueProducingEntries	decimal	Final number of total revenue-producing entries in the transaction.
[DataMember] [ForeignKey("Importer_Id")]	Importer	Contact	Importer.
[DataMember]	Importer_Id	long?	Importer Id.
[DataMember] [StringLength(12)]	ImporterOfRecordNumber	string	IRS# of Importer for whom the Daily statement was prepared. (importer statement only).
[NotMapped] [IgnoreDataMember]	IsPeriodic	bool	true if daily statement is periodic.
[DataMember] [InverseProperty("StatementDaily")]	LastStatuses	List<EntityLastStatus>	Last status collection.
[DataMember] [Index(false)]	Number	long	Not used.
[DataMember] [StringLength(1)]	PaymentType	string	Type of payment. "2", "3", "5" for non-periodic and "6", "7", "8" for periodic.

			Has Possible Values:
[DataMember] [StringLength(6)]	PaymentUnitNumber	string	Payer's Unit Number
[DataMember] [ForeignKey("PortOfEntries_Id")]	PortOfEntries	CustomsPort	District/port of Entry Summary.
[DataMember]	PortOfEntries_Id	long?	District/port Id.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryAmountDue	decimal	Preliminary Total Amount Due.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryAntidumpingDuty	decimal	Preliminary total antidumping duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryCountervailingDuty	decimal	Preliminary total countervailing duty amount.
[DataMember] [Column(TypeName = "date")]	PreliminaryDate	DateTime	The date when daily statement is imposed by Customs. Usually 8-th working day from release.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryDeferredTax	decimal	Preliminary total deferred tax amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryEstimatedDuty	decimal	Preliminary total estimated duty amount.
[IgnoreDataMember] [NotMapped] [DecimalPrecisionAttribute(11, 2)]	PreliminaryEstimatedEntriesTax	decimal	Not used.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryEstimatedTax	decimal	Preliminary total estimated tax amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryInterestAmountReconciliationSummary	decimal	Preliminary total interest amount for reconciliation summary.
[DataMember] [DecimalPrecisionAttribute(5, 0)]	PreliminaryNumberNonRevenueProducingEntries	decimal	Preliminary total number of non-revenue producing entries in the transaction.
[DataMember] [DecimalPrecisionAttribute(5, 0)]	PreliminaryNumberRevenueProducingEntries	decimal	Preliminary number of total revenue-producing entries in the transaction.
[DataMember] [StringLength(10)]	StatementMonthlyNumber	string	The number of monthly statement the daily statement belongs to. (For periodic daily statements only).
[DataMember] [StringLength(10)]	StatementNumber	string	Daily statement number.
[DataMember] [StringLength(2)]	StatementPaymentType	string	"01" – periodic daily statement, "02" – non-periodic daily statement. For future usage: "03" – Deferred Tax Bill, "04" – single pay entry summary, "05" - Post Summary Correction.

			Has Possible Values.
[DataMember] [InverseProperty("StatementDaily")] [Obsolete("LastStatuses", false)]	Statuses	List<EntityStatus>	Obsolete. Use LastStatuses instead. Statuses of Customs entry.

Interface: [ServiceContract]
SMS.Broker.ServiceContracts.Documents.IStatementDailyManager :
IEntityManagerDocument<StatementDaily> [1]

Methods:

Attributes	Name	Type	Description
[OperationContract] [FaultContract(typeof(ActionFault))]	GetByStatementNumber(string statementNumber)	StatementDaily	Returns daily statement by its StatementNumber.
[OperationContract] [FaultContract(typeof(ActionFault))]	GetReport_PeriodicDailyACHStatement(long Id, Broker.DataContracts.Reports.ReportOutputFormat output)	byte[]	returns a daily statement document as a report.
[OperationContract] [FaultContract(typeof(ActionFault))]	GetStatementsEntriesDutyAndFeePaymentInfo(long[] ids)	List<StatementEntriesDutyAndFeePaymentInfo>	Gets entries duty and fee information.
[OperationContract] [FaultContract(typeof(ActionFault))]	GetVoucherByStatementNumber(string statementNumber)	Voucher	Gets a voucher by daily statement number. (For non-periodic daily statements only).
[OperationContract(IsOneWay = true)]	PutToQUEUEOneWay(long id)	void	Put an RM-app transmission (statement Debit Authorization) to transmission queue. [2].
[OperationContract] [FaultContract(typeof(ActionFault))] [FaultContract(typeof(EntryValidationFault))]	VoucherProcessing(long id)	void	Generates a voucher by daily statement Id. (For non-periodic daily statements only).

[1] - The features of IStatementDailyManager those are different from other Operation Contract inherited classes ones.

a) New(long? SourceId) method is for future usage only. Will be applied for Single Payment and PSC daily statements when Customs put them into production.

b) Save(IStatementDailyManager entity) method is for future usage only. Will be applied for Single Payment and PSC daily statements when Customs put them into production.

c) ValidateTreeById(long Id, string action) method is for future usage only. Will be applied for Single Payment and PSC daily statements when Customs put them into production.

d) The follow statuses are used for StatementDaily document.

Entity status type:

[EntityStatusType.ABIAppeQUE](#)
 Entity status values:
[EntityStatusValue.ABIAppePutToQUE](#)
[EntityStatusValue.ABIAppeSent](#)
[EntityStatusValue.ABIAppeRejected](#)
[EntityStatusValue.ABIAppePutToQUEError](#)
[EntityStatusValue.ABIAppeIncoming](#)

Entity status type:
[EntityStatusType.ABIAppeCustoms](#)
 Entity status values:
[EntityStatusValue.ABIAppeCustomsPreliminary](#)
[EntityStatusValue.ABIAppeCustomsFinal](#)
[EntityStatusValue.ABIAppeCustomsAuthorized](#)
[EntityStatusValue.ABIAppeCustomsDeletedInCustoms](#)

On the table below you can see all changes in [StatementDaily](#) document last status (Status Type and Status Value properties) those undergo during data exchange with Customs. The actions which suggest some workflow problem are marked in red.

Action	Status Type	Status Value
PF ←Get file (preliminary)	EntityStatusType.ABIAppeQUE	EntityStatusValue.ABIAppeIncoming
PF ←Get with error	EntityStatusType.ABIAppeQUE	EntityStatusValue.ABIAppeIncomingError
PF ←Statement added	EntityStatusType.ABIAppeCustoms	EntityStatusValue.ABIAppeCustomsPreliminary
RM Put to queue→	EntityStatusType.ABIAppeQUE	EntityStatusValue.ABIAppePutToQUE
RM Transmit→	EntityStatusType.ABIAppeQUE	EntityStatusValue.ABIAppeSent
RM ←Put to queue error	EntityStatusType.ABIAppeQUE	EntityStatusValue.ABIAppePutToQUEError
PZ ←Rejected (authorization/negation rejected)	EntityStatusType.ABIAppeQUE	EntityStatusValue.ABIAppeRejected
PZ ←Accepted (authorization)	EntityStatusType.ABIAppeCustoms	EntityStatusValue.ABIAppeCustomsAuthorized
PZ ←Accepted (negation)	EntityStatusType.ABIAppeCustoms	EntityStatusValue.ABIAppeCustomsPreliminary
SQ ←Deleted (Canceled) by Customs by importer's request.	EntityStatusType.ABIAppeCustoms	EntityStatusValue.ABIAppeCustomsDeletedInCustoms
PF ←Statement status changed to final.	EntityStatusType.ABIAppeCustoms	EntityStatusValue.ABIAppeCustomsFinal

PF – Preliminary/final daily statement outgoing (our incoming) file.
 RM – ACH Debit Authorization/Entry Summary Presentation Customs incoming (our outgoing) file.
 PZ –ACH Debit Authorization/Entry Summary Presentation Customs outgoing (our incoming) file.
 SQ –ACH Update Customs outgoing (our incoming) file.

[2] – Methods can be used instead of more general method `PutToQueueOneWay(string AppId, List<Parameter> parameters)` described in `SMS.Broker.ServiceContracts.ISmsABIManager`.
 Since it is used for daily statement only it has more simple set of parameters.

`PutToQueueOneWay(long id)` - asynchronous method.

Parameters:
 id – Daily Statement Id

For example:

```

var mngr =
ClientContext.ServicesFactory.GetManager<SMS.Broker.ServiceContracts.Documents.
IStatementDailyManager>();
mngr.PutToQUEOneWay(5L);

```

2.1 Daily Statement Entry.

Class: [DataContract]

SMS.Broker.DataContracts.Documents.StatementDailyEntry : EntityChild

The class contains an information about entry placed to statement detail. Its object reflects a relation between StatementDaily object and CustomsEntry one.

Properties:

Attributes	Name	Type	Description
[DataMember] [DecimalPrecisionAttribute(11, 2)]	AntidumpingDutyAmount	decimal	Total antidumping duty amount.
[DataMember] [StringLength(1)]	AntidumpingIndicator	string	Antidumping Indicator. A code of Y (yes) indicates payment is due.
[DataMember] [StringLength(9)]	BrokerReferenceNumber	string	Broker Reference Number. The code is provided by user for Entry Summary.
[DataMember] [StringLength(2)]	ClientBranchDesignation	string	Client Branch Indicator.
[DataMember] [StringLength(1)]	ConsolidatedIndicator	string	Consolidated Indicator. A code of C indicating the entry is consolidated.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	CountervailingDutyAmount	decimal	Total countervailing duty amount.
[DataMember] [StringLength(1)]	CountervailingIndicator	string	Countervailing Indicator. A code of Y (yes) indicates payment is due.
[ForeignKey("CustomsEntry_Id")] [DataMember]	CustomsEntry	CustomsEntry	Customs Entry the info about which is in this entry info object.
[DataMember]	CustomsEntry_Id	long?	Customs Entry Id.
[DataMember] [StringLength(1)]	DeferredTaxIndicator	string	Deferred Tax Indicator. A code of Y (yes) indicates the tax is deferred.
[DataMember] [StringLength(3)]	DeleteSource	string	Has Possible Values.
[DataMember] [StringLength(3)]	EntryFilerCode	string	Entry Filer Code.
[DataMember] [StringLength(8)]	EntryNumber	string	Entry Number.
[DataMember] [StringLength(2)]	EntryType	string	Entry type.

[DataMember] [DecimalPrecisionAttribute(11, 2)]	EstimatedDutyAmount	decimal	Total duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	EstimatedTaxAmount	decimal	Total tax amount.
[DataMember] [InverseProperty("StatementDailyEntry")]	FeeAmounts	List<StatementDailyEntryFee>	The collection of entry fees.
[DataMember]	Final	bool	true if related final daily statement has come.
[IgnoreDataMember] [NotMapped]	Importer	Contact	Importer.
[DataMember] [StringLength(12)]	ImporterOfRecordNumber	string	IRS# of the importer for whom the daily statement was prepared.
[DataMember] [DecimalPrecisionAttribute(8, 2)]	InterestAmountForReconciliationSummary	decimal	Interest amount for reconciliation summary.
[DataMember] [StringLength(1)]	PayIndicator	string	Pay indicator. A code of Y (yes) indicates payment is due.
[DataMember] [StringLength(1)]	PaymentType	string	Type of payment. "2", "3", "5" for non-periodic and "6", "7", "8" for periodic. Has Possible Values:
[DataMember] [StringLength(4)]	PortOfEntry	string	A code representing the district/port of Entry Summary.
[DataMember]	ReconciliationEntry_Id	long?	Reconciliation Entry Id.
[ForeignKey("ReconciliationEntry_Id")] [DataMember]	ReconEntry	ReconEntry	Reconciliation Entry the info about which is in the entry info object.
[ForeignKey("StatementDaily_Id")] [DataMember]	StatementDaily	StatementDaily	The reference to daily statement document this daily statement entry (entry info) object belongs to.
[DataMember]	StatementDaily_Id	long	Daily statement Id.
[DataMember] [StringLength(3)]	TeamNumber	string	A number of CBP specialist team.
[IgnoreDataMember] [NotMapped] [DecimalPrecisionAttribute(11, 2)]	TotalAmountDue	decimal	Total amount due per entry that is calculated as a sum of: EstimatedDutyAmount, EstimatedTaxAmount, AntidumpingDutyAmount, CountervailingDutyAmount and FeeAmount.

Methods:

Attributes	Name	Type	Description
	ToString()	string	Returns text representation.

2.1.1 Daily Statement Entry Fee.

Class: [DataContract]

SMS.Broker.DataContracts.Documents.StatementDailyEntryFee : EntityChild

Represents an information about statement entry fee.

Properties:

Attributes	Name	Type	Description
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FeeAmount	decimal	Fee Amount.
[DataMember] [StringLength(3)]	FeeClassCode	string	User Fee Accounting Class Code. Has Possible Values.
[ForeignKey("StatementDailyEntry_Id")] [DataMember]	StatementDailyEntry	StatementDailyEntry	The reference to daily statement entry (entry info) object this fee object belongs to.
[DataMember]	StatementDailyEntry_Id	long	Daily statement entry Id.

2.2 Daily Statement Fee.

Class: [DataContract]

SMS.Broker.DataContracts.Documents.StatementDailyFee : EntityChild

Represents an information about statement fee.

Properties:

Attributes	Name	Type	Description
[DataMember] [StringLength(3)]	FeeClassCode	string	User Fee Accounting Class Code. Has Possible Values.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalFeeAmount	decimal	Final fee amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryFeeAmount	decimal	Preliminary fee amount.
[ForeignKey("StatementDaily_Id")] [DataMember]	StatementDaily	StatementDaily	The reference to daily statement document this fee object belongs to.
[DataMember]	StatementDaily_Id	long	Daily statement Id.

3. Monthly Statement.

Monthly Statement document represents an info about duties and fees owed to CBP for the entries.

Class: [DataContract](#)

SMS.Broker.DataContracts.Documents.[StatementMonthly](#) : [EntityDocument](#)

Header data of Monthly Statement document.

Properties:

Attributes	Name	Type	Description
[DataMember] [InverseProperty("StatementMonthly")]	CustomsEntries	List<StatementMonthlyEntry>	The monthly statement entry (entry info) collection.
[DataMember] [Column(TypeName = "date")]	DueDate	DateTime?	Monthly statement due date.
[DataMember] [InverseProperty("StatementMonthly")]	FeeAmounts	List<StatementMonthlyFee>	The collection of monthly statement fees.
[DataMember] [Column(TypeName = "date")]	FinalDate	DateTime?	The date when final monthly statement is issued by Customs (usually 15-th working day of month).
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalTotalAmountDue	decimal	Final total amount due.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalTotalAntidumpingDuty	decimal	Final total antidumping duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalTotalCountervailingDuty	decimal	Final total countervailing duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalTotalDuty	decimal	Final total duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalTotalTax	decimal	Final Total Tax.
[DataMember] [ForeignKey("Importer_Id")]	Importer	Contact	Importer.
[DataMember]	Importer_Id	long?	Importer Id.
[DataMember] [StringLength(12)]	ImporterOfRecordNumber	string	IRS# of the importer for whom the monthly statement was prepared. (For importer statement only).
[DataMember] [InverseProperty("StatementMonthly")]	LastStatuses	List<EntityLastStatus>	Last status collection.
[DataMember] [Index(false)]	Number	long	Not used.
[DataMember] [StringLength(1)]	PaymentType	string	"6", "7", "8" periodic payment type. Has Possible Values.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryTotalAmountDue	decimal	Preliminary total amount due.

2)]			
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryTotalAntidumpingDuty	decimal	Preliminary total antidumping duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryTotalCountervailingDuty	decimal	Preliminary total countervailing duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryTotalDuty	decimal	Preliminary total duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryTotalTax	decimal	Preliminary Total Tax.
[DataMember] [Column(TypeName = "date")]	PrintDate	DateTime	The date when preliminary monthly statement is issued by Customs (usually 11-th working day of month).
[DataMember] [StringLength(10)]	StatementNumber	string	Periodic monthly statement number.
[DataMember] [InverseProperty("StatementMonthly")]	Statements	List<StatementMonthlyStatementDaily>	Daily statements collection.
[DataMember] [InverseProperty("StatementMonthly")] [Obsolete("LastStatuses", false)]	Statuses	List<EntityStatuses>	Obsolete. Use LastStatuses instead. Statuses of Customs entry.

Interface: [ServiceContract]
SMS.Broker.ServiceContracts.Documents.IStatementMonthlyManager :
IEntityManagerDocument<StatementMonthly> [1]

Methods:

Attributes	Name	Type	Description
[OperationContract] [FaultContract(typeof(ActionFault))]	GetByStatementNumber(string statementNumber)	StatementMonthly	Returns monthly statement by its StatementNumber.
[OperationContract] [FaultContract(typeof(ActionFault))]	GetStatementsByStatementMonthlyNumber(string statementNumber)	List<StatementMonthlyStatementDaily>	Returns a collection of daily statements by monthly StatementNumber.
[OperationContract] [FaultContract(typeof(ActionFault))]	GetVoucherByStatementNumber(string statementNumber)	Voucher	Gets a voucher by monthly statement number.
[OperationContract] [FaultContract(typeof(ActionFault))] [FaultContract(typeof(EntryValidationFault))]	VoucherProcessing(long id)	void	Generates a voucher by monthly statement Id.

1] - The features of IStatementMonthlyManager those are different from other Operation Contract inherited classes.

a) New(long? SourceId) method is not applicable because monthly statement always appears on Customs side first and then added automatically to SMS-database.

b) Save(IStatementDailyManager entity) method is not applicable because monthly statement always appears on Customs side first and then added automatically to SMS-database.

c) `ValidateTreeById(long Id, string action)` method is not applicable because monthly statement always appears on Customs side first and then added automatically to SMS-database.

d) The follow statuses are used for `StatementMonthly` document.

Entity status type:

`EntityType.ABIAppQUE`

Entity status values:

`EntityStatusValue.ABIAppIncoming`

Entity status type:

`EntityType.ABIAppCustoms`

Entity status values:

`EntityStatusValue.ABIAppCustomsPreliminary`

`EntityStatusValue.ABIAppCustomsFinal`

On the table bellow you can see all changes in `StatementMonthly` document last status (Status Type and Status Value properties) those undergo during data exchange with Customs. The actions which suggest some workflow problem are marked in red.

Action	Status Type	Status Value
MS ←Get file (preliminary)	<code>EntityType.ABIAppQUE</code>	<code>EntityStatusValue.ABIAppIncoming</code>
MS ←Get with error	<code>EntityType.ABIAppQUE</code>	<code>EntityStatusValue.ABIAppIncomingError</code>
MS ←Statement added	<code>EntityType.ABIAppCustoms</code>	<code>EntityStatusValue.ABIAppCustomsPreliminary</code>
MS ←Statement status changed to final.	<code>EntityType.ABIAppCustoms</code>	<code>EntityStatusValue.ABIAppCustomsFinal</code>

MS – Preliminary/final daily statement outgoing (our incoming) file.

3.1 Monthly Statement Entry.

Class: `[DataContract]`

`SMS.Broker.DataContracts.Documents.StatementMonthlyEntry : EntityChild`

The class contains an information about entry placed to daily statement that belongs to the given monthly statement.

Properties:

Attributes	Name	Type	Description
<code>[DataMember]</code> <code>[StringLength(3)]</code>	DeleteSource	<code>string</code>	A code representing the source of the delete. Has Possible Values.
<code>[DataMember]</code> <code>[StringLength(3)]</code>	EntryFilerCode	<code>string</code>	Entry filer code.
<code>[DataMember]</code> <code>[StringLength(8)]</code>	EntryNumber	<code>string</code>	Entry number.
<code>[DataMember]</code> <code>[StringLength(10)]</code>	PeriodicDailyStatementNumber	<code>string</code>	StatementNumber of periodic daily statement this monthly statement

			entry (entry info) object belongs to.
[ForeignKey("StatementMonthly_Id")] [DataMember]	StatementMonthly	StatementMonthly	The reference to monthly statement document this monthly statement entry (entry info) object belongs to.
[DataMember]	StatementMonthly_Id	long	Monthly statement Id.

Methods:

Attributes	Name	Type	Description
	ToString()	string	Returns text representation.

3.2 Monthly Statement Fee.

Class: [DataContract]

SMS.Broker.DataContracts.Documents.StatementMonthlyFee : EntityChild

Represents an information about monthly statement fee.

Properties:

Attributes	Name	Type	Description
[DataMember] [StringLength(3)]	FeeClassCode	string	User Fee Accounting Class Code. Has Possible Values.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	FinalFeeAmount	decimal	Final fee amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	PreliminaryFeeAmount	decimal	Preliminary fee amount.
[ForeignKey("StatementMonthly_Id")] [DataMember]	StatementMonthly	StatementMonthly	The reference to monthly statement document this fee object belongs to.
[DataMember]	StatementMonthly_Id	long	Monthly statement Id.

3.3 Daily Statement of Monthly Statement.

Class: [DataContract]

SMS.Broker.DataContracts.Documents.StatementMonthlyStatementDaily : EntityChild

The class contains an information about daily statement placed to monthly statement detail. Its object reflects a relation between StatementMonthly object and StatementDaily one.

Properties:

Attributes	Name	Type	Description
[DataMember] [Column(TypeName = "date")]	EntrySummaryPresentationDate	DateTime	The day when statement Debit Authorization transaction (RM-app) was processed by CBP.
[DataMember] [StringLength(3)]	FilerCode	string	Filer code.
[DataMember] [StringLength(12)]	ImporterNumber	string	IRS# of importer for whom the periodic daily statement was prepared (importer statement only).
[DataMember] [StringLength(4)]	Port	string	A code representing the district/port for which the periodic daily statement was prepared.
[ForeignKey("Statement_Id")] [DataMember]	Statement	StatementDaily	Daily statement the info about which is in this daily statement info object.
[DataMember]	Statement_Id	long?	Daily statement Id.
[ForeignKey("StatementMonthly_Id")] [DataMember]	StatementMonthly	StatementMonthly	The reference to monthly statement document this daily statement (daily statement info) object belongs to.
[DataMember]	StatementMonthly_Id	long	Monthly statement Id.
[DataMember] [StringLength(10)]	StatementNumber	string	Periodic daily StatementNumber.
[DataMember] [Column(TypeName = "date")]	StatementPrintDate	DateTime	Preliminary periodic daily statement date.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	TotalAmountDue	decimal	Total amount due.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	TotalAntidumpingDuty	decimal	Total antidumping duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	TotalCountervailingDuty	decimal	Total countervailing duty.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	TotalDuty	decimal	Total duty amount.
[DataMember] [DecimalPrecisionAttribute(11, 2)]	TotalTax	decimal	Total tax amount.

Methods:

Attributes	Name	Type	Description
	ToString()	string	Returns text representation <code>base.ToString() + " " + StatementNumber</code>