

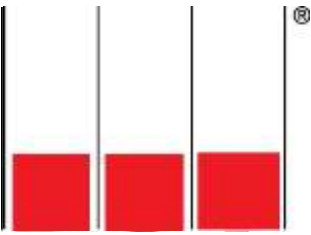
4. Statement Class

Version: 1.5

Date: 20 August 2016

Author: Gavin Ireland FISTC

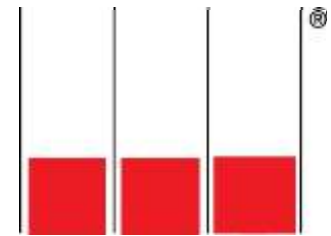
All copyright and design rights in written and electronic material produced by us for you remains ours unless and until any agreement is reached for you to use it. Any unauthorised use may constitute an infringement of copyright or design right which would be actionable in the courts.



Contents

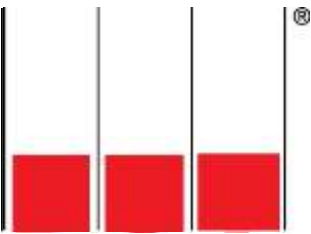
1	<i>Record of amendments</i>	3
2	<i>Fast statement class</i>	4
2.1	Product item class	13
2.2	FundCollection Class – contains a list of Fund Items	16
2.3	FundItem Class – contains a list of Transaction Collections.....	18
2.4	TransactionCollection Class.....	21
2.5	TransactionItems Class.....	23
3	<i>Transaction types</i>	26
4	<i>Contact details</i>	27
5	<i>Appendix</i>	28
5.1	Mapping fields from FAST statements data to [REDACTED]	28
5.1.1	Class - FastRecord01	28
5.1.2	Class - FastRecord150	29
5.1.3	Class – FastRecord190	29

4. Statement Class



1 Record of amendments

ID	Date	V	Amendment	Reason	Elements amended	Amended by
5	10/10/2011	1.4	New template applied, content updated	Document update	All	Gavin Ireland
6	24/06/2012	1.5	DST Rebrand	Rebranding	All	Gavin Ireland



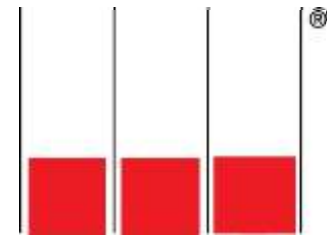
2 Fast statement class

The base Class is FastStatement.cs unless otherwise stated, all references to Classes and Properties have the '**FastStatement.**' Prefix.

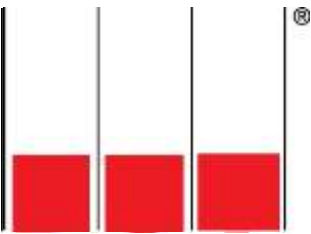
Name	Derived from	Logic
.Agent		
.Agent_YN	fastUnitholder.Record150.AGENT_YN	if fastUnitholder.Record150 is not null Agent_YN = fastUnitholder.Record150.AGENT_YN
.AgentConsolidateKey	Agent.AgentCode ClientConsolidateKey	If the Agent record is not Null add the .AgentCode and ClientConsolidateKey to the string.
.AmountInvested	ProductItem.AmountInvested	For each ProductItem add the AmountInvested to get the overall total of all products
.AmountWithdrawn	ProductItem.AmountWithdrawn	For each ProductItem add the AmountWithdrawn to get the overall total of all products
.AgentCopy	Record150.AGENT_YN FastStatement .Pull Statics.Presets.Criteria.ReferenceLookups	If Record150.AGENT_YN = "Y" and FastStatement is null and FastStatement.Pull is not true and



4. Statement Class



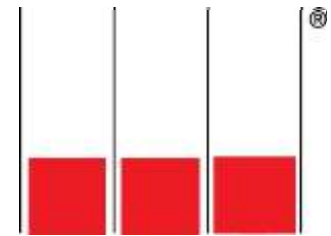
		Client Account Number is not in Agent Copy Suppress lookup set to true
.AME	FundCollection.AME	
.ASF	fastUnitholder.ASF	
.Asterix	.RecipientAddress	if .RecipientAddress contains "*" set to true
.BFValue	.ProductItem.BFValuation	For each ProductItem add the BFValue to get the overall total of all products
.ClientAccountNumber	Record01.UNHLDID	
.ClientConsolidate Key	_primaryClientName RecipientName PrimaryHolderAddress JointHolders	if Consolidate is true add _primaryclientname and .RecipientName to the string; add each line of the .PrimaryHolderAddress; and add each jointHolder from .JointHolders to the string
.ClientID	Record150.CLIENT_ID	if .Record150 is not Null ClientID = fastUnitholder.Record150.CLIENT_ID
.Consolidate	LookupEnums.NoConsolidation	if Pull Statics.Resources.Lookups[LookupEnums.NoConsolidation].Lookup.Contains(int.Parse(ClientAccountNumber)) set to true, else set to false



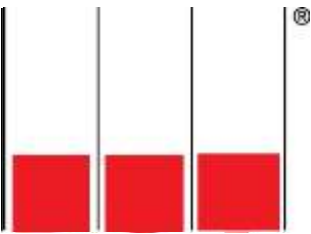
4. Statement Class

<code>.ConsolidationKey</code>	<code>.AgentConsolidateKey</code> <code>.ClientConsolidateKey</code>	if CTV is true, return "" if CTV is not true and AgentCopy is true, return <code>.AgentConsolidateKey</code> else return <code>.ClientConsolidateKey</code>
<code>.CorporateNominee</code>	<code>Record150.UNTYDS</code> <code>.RecipientAddress</code>	if <code>Record150.UNTYDS</code> contains "Corporate" or <code>.RecipientAddress</code> contains "Nominee" set to true
<code>.CTV</code>	<code>Record600List</code> <code>Record190</code>	if both of these records are populated set CTV to true
<code>.DataFileType</code>	<code>InputDataFileTypes</code>	returns the text from ██████████.Composer.InputDataFileTypes
<code>.DataID</code>	<code>.Pull</code> <code>.AgentCopy</code>	if Pull is true set to "Pulls" if AgentCopy is true set to "Agent" else set to "Client"
<code>.DDA</code>	<code>LookupEnums.DDA</code>	if <code>.Record150</code> is not null, set to <code>Statics.Resources.Lookups[LookupEnums.DDA].Lookup.Contains(int.Parse(ClientAccountNumber))</code>
<code>.Deceased</code>	<code>.RecipientAddress</code>	if <code>.RecipientAddress</code> contains "DECEASED" "DECD" "Dec'd" "Desc'd"

4. Statement Class



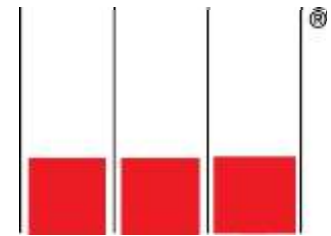
		set to true
.Denominations	.FundItem.Denomination	if FundItem.Denomination contains the following: both share and units returns "Shares/Units" share returns "Shares" units returns "Units"
.DoNotUse	.RecipientAddress	if .RecipientAddress contains "DO NOT" set to true
.FromDate	Record150.LAST_WORKING_D AY	if .Record150 is not Null, set to fastUnitholder.Record150.LAST_WORKING_DAY
.FromDateDesc	Statics.FormatFastDate	return Statics.FormatFastDate (FromDate, false, false)
.Funds	.FundCollection	A list of Fund Items
.GoneAway	.RecipientAddress	if .RecipientAddress contains "GONEAWAY" "GONE AWAY" set to true
.HasMandGFunds	Funds.HasMandGFunds	return Funds.Has █████ Funds
.HasPruFunds	Funds.HasPruFunds	return Funds.Has █████ Funds
.IncomeDistributed	ProductItem.IncomeDistributed	For each ProductItem add the IncomeDistributed to get the overall total of all products



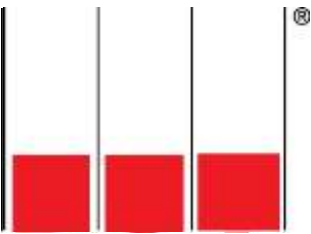
4. Statement Class

<code>.IncomeReinvested</code>	<code>ProductItem.IncomeReinvested</code>	For each ProductItem add the IncomeReinvested to get the overall total of all products
<code>.ISAAnnualNotice</code>	<code>.AmountInvested</code> <code>.SharedMembers.FastToDate.Substring</code>	if (<code>newProduct.AmountInvested > 0.00</code> && <code>Statics.SharedMembers.FastToDate.Substring(4, 2) == "04"</code>) set to true
<code>.IsDirect</code>	<code>Agent_YN</code>	if <code>Agent_YN</code> is not "Y" set to true
<code>.LiquidatedCount</code>	<code>Funds.Items</code> <code>Transactions.Items</code>	Starting at 0, for each Item in <code>.Funds.Items</code> and each trans in <code>.Transactions.Items</code> add 1 to the count
<code>.CashISA</code>	<code>product.MaxiCashISA</code>	Loop through all of the product items in <code>Products</code> and if <code>product.CashISA</code> is not Null set to true
<code>.MCID</code>	<code>fastUnitholder.Record150.MCID</code>	if <code>fastUnitholder.Record150</code> is not Null <code>MCID = fastUnitholder.Record150.MCID</code>
<code>.MGPull</code>	<code>LookupEnums.Pulls</code>	if <code>fastUnitholder.Record150</code> is not Null <code>Statics.Resources.Lookups[LookupEnums.MGPulls].Lookup.Contains(int.Parse(Client AccountNumber))</code>

4. Statement Class



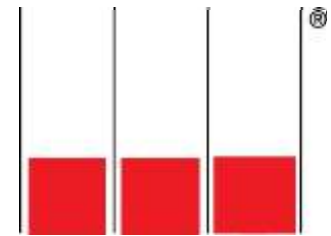
.Nil		if fastUnitholder.Record150 is not Null (Valuation == 0)
.NoProduct	Record540	if there are no products within the 540 record set to true
.NullDistributions	Record560	if .NoProduct is true NullDistributions = fastUnitholder.Record560List
.Pull	CorporateNominee Asterisk TwoLineAddress Deceased GoneAway DoNotUse	if (.CorporateNominee .Asterisk .TwoLineAddress .Deceased .GoneAway .DoNotUse) then Pull is true
.HasDesignation	.ProductItem.Designation	is true if ProductItem.Designation.Length > 0
.IF7	FundCollection.IF7	
.ISA		Boolean – initiated to false Set to true if .ProductType == ISA
.JointHolders	Record150.JOINT_CLIENT1 Record150.JOINT_CLIENT2 Record150.JOINT_CLIENT3 Record150.JOINT_CLIENT4	



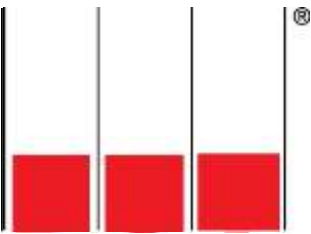
4. Statement Class

.MultiAccounts		Set to true when adding FastStatement Object if .ClientAccountNumber != FastStatement.ClientAccountNumber set to true
.NonMarketable	.ClientAccountNumber	if the .ClientAccountNumber is matched to and external lookup file – NonMarketable.txt (See Getting Started documentation) or the Agent record is present set to true
.PEP		Boolean – initiated to false Set to true if .ProductType == PEP
.primaryclientaddress1	Record150.PRIMARY_CLIENT_ADDRL1	if fastUnitholder.Record150 is not Null PrimaryClientAddress1 = fastUnitholder.Record150.PRIMARY_CLIENT_ADDRL1
.PrimaryHolderName	Record150.PRIMARY_CLIENT_NAME	
.PrimaryHolderAddress	Record150.PRIMARY_CLIENT_ADDRL1 Record150.PRIMARY_CLIENT_ADDRL2 Record150.PRIMARY_CLIENT_ADDRL3 Record150.PRIMARY_CLIENT_ADDRL4	Nexdax Address object

4. Statement Class



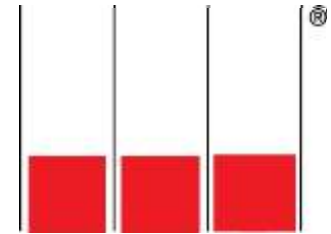
	Record150.PRIMARY_CLIENT_ADDRL5 Record150.PRIMARY_CLIENT_POSTCDE	
.PrimaryHolderPostcode	Record150.PRIMARY_CLIENT_POSTCDE	
.Products	.ProductItem	A List of Product Items
.RecipientAddress	Record01.RECIPIENT_ADDRL1 Record01.RECIPIENT_ADDRL2 Record01.RECIPIENT_ADDRL3 Record01.RECIPIENT_ADDRL4 Record01.RECIPIENT_ADDRL5 Record01.RECIPIENT_POSTCODE	If present added to the [REDACTED] Address object
.RecipientFAO	Record150.RECIPIENT_FOR_ATTENTION	
.RecipientName	Record01.RECIPIENT_NAME	
.RecipientPostcode	Record01.RECIPIENT_POSTCODE	
.RecipientSalutation	Record150.RECIPIENT_SALUTATION	



4. Statement Class

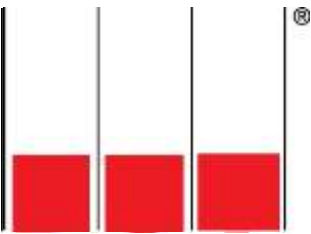
<code>.RunDate</code>	Record150.RRUNDTE	if .Record150 is not Null RunDate = fastUnitholder.Record150.RRUNDTE
<code>.SavingsPlan</code>		Boolean – initiated to false Set to true if .ProductType == SavingsPlan
<code>.SummaryEndPage</code>	_summaryEndPage	if 6- _summaryEndPage is greater than 1, return “pages 6-“ else return “page “
<code>.ToDate</code>	Record150.TRNTODT	if .Record150 is not Null ToDate = fastUnitholder.Record150.TRNTODT
<code>.ToDateDesc</code>	Statics.FormatFastDate	return Statics.FormatFastDate (ToDate, true, false)
<code>.TotalCashISA</code>	.ProductItem. CashISA.Valuation	If a fund is within the Cash ISA product, add the valuation for each fund and represent the overall value of the Cash ISA
<code>.TwoLineAddress</code>	.RecipientAddress	if .RecipientAddress contains 2 lines or less set to true
<code>.USInvestor</code>	LookupEnums.USClients	if .Record150 is not Null USInvestor = Statics.Resources.Lookups[LookupEnums.USClients].Lookup.Contains(int.Parse(ClientAccountNumber))
<code>.Valuation</code>	.FundItem.Valuation	For each fund, add the fund.valuation to obtain the overall total

4. Statement Class



2.1 Product item class

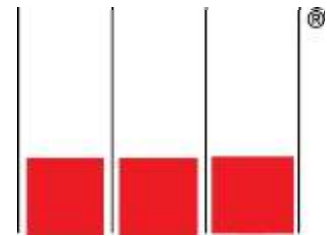
Name	Derived from	Logic
.AccountNumber	.Record540.UNHLDID	
.AdditionalSubscription	.Record540.TAX_YEAR .Record540.ASF_TYPE_FLAG	if (.AmountStillSubscribe >= 100.0 && .Record540.TAX_YEAR.Contains("2008") && .Record540.ASF_TYPE_FLAG == "C")
.AmountInvested	FundCollection.AmountInvested	Matches Product Code and Account Number to get the Amount Invested of a product for all the funds in that collection
.AmountSubscribed	.Record540.EQUITY_INVESTED .Record540.CASH_INVESTED	Add both these figures to return the amount subscribed (EQUITY_INVESTED + CASH_INVESTED)
.AmountStillSubscribed	.OverallLimit .AmountSubscribed	minus AmountSubscribed from the OverallLimit to get the amount still subscribed (.OverallLimit – .AmountSubscribed)
.AmountWithdrawn	FundCollection.AmountWithdrawn	Matches Product Code and Account Number to get the Amount Withdrawn of a product for all the funds in that collection
.Valuation	FundCollection. Valuation	Matches Product Code and Account Number to get the Valuation of a product for all the funds in that collection



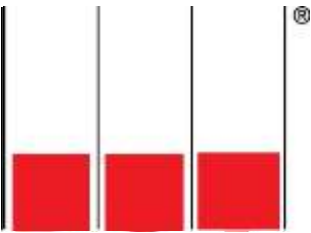
4. Statement Class

.Designation	.Record150.DESIGNATION	
.FromDate	.Record150.LAST_WORKING_DAY	
.IncomeDistributed	FundCollection.IncomeDistributed	Matches Product Code and Account Number to get the Income Distributed amount of a product for all the funds in that collection
.IncomeReinvested	FundCollection.IncomeReinvested	Matches Product Code and Account Number to get the Income Reinvested amount of a product for all the funds in that collection
.████CashISA	FundCollection.████CashISA	Matches Product Code and Account Number to get the Total value of the █████ Cash ISA product for all the funds in that collection
.OverallLimit		Hard coded to 7200 because of data issue
.ProductCode	.Record540.PRODUCT_CODE	
.ProductCodeSeq	.rec540.PRODUCT_CODE .rec540.PHI_SEQ	ProductCodeSeq = int.Parse (rec540.PRODUCT_CODE + rec540.PHI_SEQ)
.ProductSeq	.rec540.PHI_SEQ	ProductSeq = int.Parse (rec540.PHI_SEQ)
.ProductType	.ProductCode	Takes the product code parameter and matches the code to a name: 1: █████ 4: .████

4. Statement Class



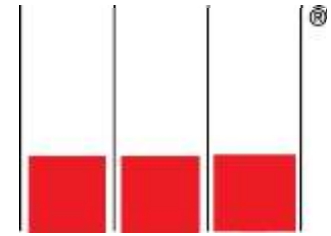
		7: [REDACTED] 8: [REDACTED] 2: [REDACTED]
.ProductName	.Record540.SHORT_PRODUCT_NAME	
.StockTransferIn	FundCollection.StockTransferIn	Matches Product Code and Account Number to get the stock transfer in of a product for all the funds in that collection
.StockTransferOut	FundCollection.StockTransferOut	Matches Product Code and Account Number to get the stock transfer out of a product for all the funds in that collection
.TaxYearEnd	.Record540.TAX_YEAR_END	
.ToDate	.Record150.TRNTODT	
.TotalIncome	FundCollection.TotalIncome	Matches Product Code and Account Number to get the total income of a product for all the funds in that collection
.TransferIn	FundCollection.TransferIn	Matches Product Code and Account Number to get the transfer in of a product for all the funds in that collection
.TransferOut	FundCollection.TransferOut	Matches Product Code and Account Number to get the transfer out of a product for all the funds in that collection
.Valuation	FundCollection.Valuation	Matches Product Code and Account Number to get the Valuation of a product for all the funds in that collection



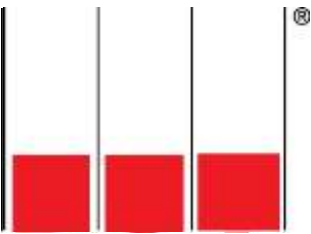
2.2 FundCollection Class – contains a list of Fund Items

Name	Derived from	Logic
.AME	FundItem.FundCode	is set to true if FundItem.FundCode matches: "██████" "██████" "██████" "██████"
.AmountInvested	FundItem.TransactionsCollection.AmountInvested	Match the product code and account number to FundItem Transactions List and add up all the amounts invested See Transaction Collection for Amount Invested Logic
.AmountWithdrawn	FundItem.TransactionsCollection.AmountWithdrawn	Match the product code and account number to FundItem Transactions List and add up all the amounts withdrawn See Transaction Collection for Withdrawal Logic
.██████Valuation	FundItem.██████Valuation	Matches Product Code and Account Number to get the valuation of a Product for all the funds in that collection
.AmountWithdrawn	FundItem.TransactionsCollection.AmountWithdrawn	Matches Product Code and Account Number to FundItem List and add up all the amount withdrawn

4. Statement Class



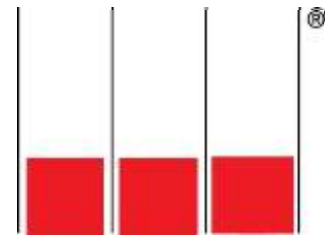
.Has████ Funds	FundItem.Is████	Loops through the FundItem list and returns true if a fund has a █████
.Has████ Funds	FundItem.Is████	Loops through the FundItem list and returns true if a fund has a █████
.HasRollOver	FundItem.HasRollOver	Loops through the FundItem list and returns true if a fund has a HasRollOver
████	FundItem.SubFundNoteID	is set to true if FundItem.SubFundNoteID == 7
.IncomeDistibuted	FundItem.TransactionsCollection.IncomeDistributed	Match the product code and account number to FundItem Transactions List and add up all the amounts distributed See Transaction Collection for Distribution Logic
.IncomeReinvested	FundItem.TransactionsCollection.IncomeReinvested	Match the product code and account number to FundItem Transactions List and add up all the amounts reinvested See Transaction Collection for Withdrawal Logic
████CashISA	FundItem.████CashISA	Match the product code and account number to the FundItem List and return the all the funds that are part of the █████ Cash ISA product – Then use this collection of █████CashISA funds to calculate amounts etc.



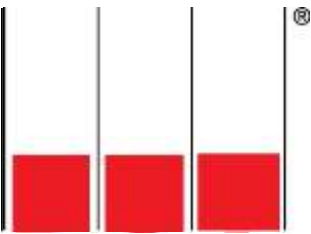
2.3 FundItem Class – contains a list of Transaction Collections

Name	Derived from	Logic
.AccountNumber	FastRecordFund.UNITHOLDER_ID	
.BFUnits	FastRecordFund.BROUGHT_FORWARD_UNITS	
.BFValuation	FastRecordFund.BF_VALUE	
.Denomination	FastRecordFund.DENOMINATION	if .Denomination contains the following: both share and units returns “Shares/Units” share returns “Shares” units returns “Units”
.Designation	FastRecordFund.DESIGNATION	
.FundCode	FastRecordFund.SHORT_FUND_ID	
.FundName	FastRecordFund.FUNDNAME_LNG	
.FundSubName		Takes Fund code and fund long name and matches this to the data contained in the FundDetailItem Class populated from FundDetailsFileXLS (an excel file) which is named in the ████████ Constants config file.
.HasRollOver	FundItem.Transactions	Loops through the TransactionItem list and returns true if

4. Statement Class



		a fund has a transaction
.HideFund	FundItem.FundCode FundItem. Valuation FundItem.Valuation FundItem.Transactions.Count	if FundCode is "CI1" set to true if Valuation, Valuation and Transactions.Count are all zero set to true
.Is Trust	.FundCode	Is set to true if the fundcode equals the following codes: "EIT"
.IsHighIncomeInvestmentTrust	.FundCode	Is set to true if the fundcode equals the following codes: "HII" "HIP"
.IsIncomeInvestmentCompany	.FundCode	Is set to true if the fundcode equals the following codes: "IIC"
.IsInvestmentTrust	.FundCode	Is set to true if the fundcode equals the following codes: "IIT" "RIT" "HII" "HIP" "IIC" "RIC" "EIT"
.Is	.FundName	If .FundName does not contain " " set to true
.IsPropertyPF	.FundCode	Is set to true if the fundcode equals the following codes: "PAI" "PGM" "PRI" "PSI" "PAM"
.Is	.FundName	If .FundName contains " " set to true
.IsRecoveryInvestmentCompany	.FundCode	Is set to true if the fundcode equals the following codes: "RIC"

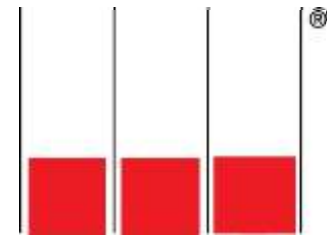


4. Statement Class

.CashISA	.FundCode	If .FundCode == C11 then CashISA is true
.PPf	IsPropertyPF(FundCode)	
.Price	FastRecordFund.VALUATION_PRICE	
.ProductCode	FastRecordFund.PRODCDE	
.ProductName	FastRecordFund.SHORT_PRODUCT_NAME	
.SubFundNotelD	.FundCode FastRecordFund.FUNDNAME_LNG	Takes Fund code and fund long name and matches this to details gathered from FundDetailsFileXLS (an excel file) - using the FundDetailCollection Class. This is then used in an ltf file. Both are named in the Constants config file.
.TableType	FastRecordFund.FUNDNAME_LNG	TableType = Statics.FundDetails[FundCode, fund.FUNDNAME_LNG].GetTableType(FundCode)
.Transaction	FundItem.FundSubname	Transactions = new TransactionCollection(fund, fastDataRecord, FundSubname.ToUpper().Contains("ACCUMULATION"), Statics.GetProductType (ProductCode))
.Units	FastRecordFund.VALUATION_UNITS	
.Valuation	FastRecordFund.VALUATION	

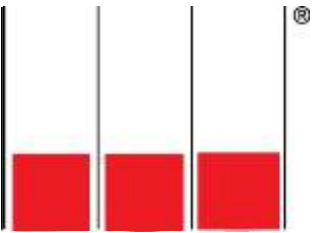


4. Statement Class



2.4 TransactionCollection Class

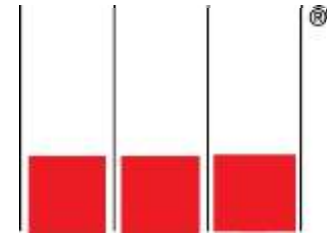
Name	Derived from	Logic
.AmountInvested	.TransactionItem.Consideration	if <code>TransactionItem.TransactionType == 1 or 11 or 26</code> Then Add <code>TransactionItem.Consideration</code> (FastDataRecord.CONSIDERATION) See Chapter 6 – Transaction Types
.AmountWithdrawn	.TransactionItem.NetAllocation	if <code>TransactionItem.TransactionType == 2 or 18 or 27</code> Then Add <code>TransactionItem.NetAllocation</code> (FastDataRecord.NET_ALLOCATION)
.Count	.Items	Gets the number of elements contained in the TransactionItem list
.IncomeDistributed	.TransactionItem.Distribution	if <code>TransactionItem</code> is a Distribution && <code>TransactionItem.MandateMethod == 000001 or 000002</code>



4. Statement Class

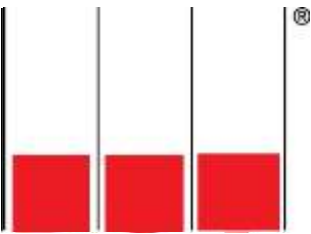
		Then Add TransactionItem.Consideration (FastRecord560.MANDATE_METHOD) (FastDataRecord.CONSIDERATION)
.IncomeReinvested	.TransactionItem.IncomeReinvested	if TransactionItem.TransactionType == 9 or 10 or 31 or 32 Then Add TransactionItem.Consideration (FastDataRecord.CONSIDERATION)
.LiquidatedCount	TransactionCollection.Items	Starting at 0, for each trans in TransactionCollection.Items add 1 to the count
.note	.TransactionItem. note	
.TotalIncome	.IncomeReinvested .IncomeDistibuted .TransactionItem.Consideration	For each Transaction Item: If .TransactionType == 36 Add the following: IncomeReinvested + IncomeDistibuted + TransactionItem.Consideration otherwise just add IncomeReinvested + IncomeDistibuted

4. Statement Class



2.5 TransactionItems Class

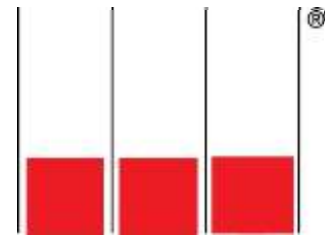
Name	Derived from	Logic
.Amount	FastRecordTransaction.CONSIDERATION	
.AmountInvested	FastRecordTransaction.NET_ALLOCATION	
.AmountPaid	FastRecordTransaction.CASH_AMOUNT_RECEIVED	
.AmountWithdrawn	FastRecordTransaction.NET_ALLOCATION	
.Consideration	FastRecordTransaction.CONSIDERATION FastRecord560.DISTRIBUTION_NET_AMT	
.Date	FastRecordTransaction.DEAL_DATE FastRecord560.DEAL_DATE	
.Discount	FastRecordTransaction.DISCOUNT_AMT	
.DiscountPercentage	FastRecordTransaction.DISCOUNT_PERCENTAGE	
.Distribution	FastRecord560	if Fast record 560 exists, then set to true
.EventTypeID	FastRecord560.EVENT_TYPE_ID	
.FundCode	FastRecordTransaction.FUND_CDE	



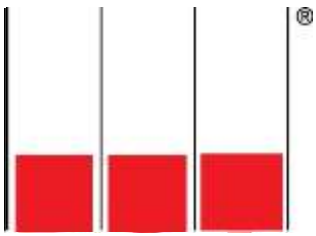
4. Statement Class

.InitialCharge	FastRecordTransaction.INITIAL_CHARGE_AMT	
.HideAmount	FastRecordTransaction.CONSIDERATION	If this Fast field contains Nil, then set to true
.HideAmountInvested	FastRecordTransaction.NET_ALLOCATION	If this Fast field contains Nil, then set to true
.HideAmountPaid	FastRecordTransaction.CASH_AMOUNT_RECIEVED	If this Fast field contains Nil, then set to true
.HideAmountWithdrawn	FastRecordTransaction.NET_ALLOCATION FastRecordTransaction.EXTCHGAMT	If either of these Fast fields contains Nil, then set to true
.HideDiscount	FastRecordTransaction.DISCOUNT_AMT	If this Fast field contains Nil, then set to true
.HideInitialCharge	FastRecordTransaction.INITIAL_CHARGE_AMT	If this Fast field contains Nil, then set to true
.HideManagementCharge	FastRecordTransaction.INITIAL_CHARGE_AMT FastRecord560.PLAN_CHARGE_AMT FastRecord560.PLAN_VAT_AMT	If this Fast field contains Nil, then set to true if it's a distribution both PLAN_CHARGE_AMT && PLAN_VAT_AMT must be Nil
.HidePrice	FastRecordTransaction.UNIT_PRICE	If this Fast field contains Nil, then set to true
.HideStampDuty	FastRecordTransaction.INITIAL_CHARGE_AMT	If this Fast field contains Nil, then set to true
.HideUnits	FastRecordTransaction.DEAL_UNITS	If this Fast field contains Nil, then set to true
.HideWithdrawalFee	FastRecordTransaction.EXTCHGAMT	If this Fast field contains Nil, then set to true

4. Statement Class



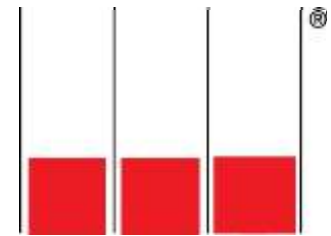
.InternalFundCode	FastRecord560.FUND_CDE	
.ManagementCharge	FastRecordTransaction.INITIAL_CHARGE_AMT	
.MandateMethod	FastRecord560.MANDATE_METHOD	
.NetAllocation	FastRecord560.NET_ALLOCATION	
.note	.FundCode .TransactionType	Is set to true if .FundCode contains "█" and .TransactionType equals one of the following: 1 5 10 11 26
.Price	FastRecordTransaction.UNIT_PRICE	
.ProductCode	FastRecord560.PRODCDE	
.Units	FastRecordTransaction.UNITS	
.ShortFundCode	fundCode	
.StampDuty	FastRecordTransaction.INITIAL_CHARGE_AMT	
.TotalPayment	FastRecord560.TOTAL_PAYMENT	
.Transaction	FastRecordTransaction.TRANSACTION_TYPE_DESCRIPTION FastRecord560.EVENT_TYPE_DESCRIPTION	
.TransactionType	FastRecordTransaction.TRANSACTION_TYPE	See Section 9 – Transaction Types



4. Statement Class

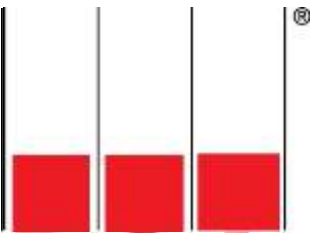
3 Transaction types

4. Statement Class



4 Contact details

Name – Project role	Role	Company - Dept	Phone	Email
Gavin Ireland - Document Author	Technical Author	DST - NexDox	[REDACTED]	[REDACTED]
[REDACTED] - Application Developers	Developer	DST - NexDox	[REDACTED]	[REDACTED]
[REDACTED] – Application Manager	Development Manager	DST - NexDox	[REDACTED]	[REDACTED]
[REDACTED]	CRM	DST – NexDox	[REDACTED]	[REDACTED]



5 Appendix

5.1 Mapping fields from FAST statements data to [REDACTED]

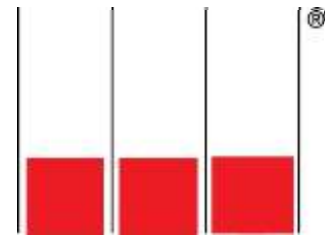
5.1.1 Class - FastRecord01

using Fast Record RQ 00686C (Record ID 01)

Fast Name – Header Record

Access Modifier	Data Type	Name	Fast Mapping (position, length)
public	string	RECIPIENT_NAME	(103, 60)
public	string	RECIPIENT_ADDRL1	(163, 40)
public	string	RECIPIENT_ADDRL2	(203, 40)
public	string	RECIPIENT_ADDRL3	(243, 40)
public	string	RECIPIENT_ADDRL4	(283,40)
public	string	RECIPIENT_ADDRL5	(323, 40)
public	string	RECIPIENT_POSTCDE	(363, 10)
public	string	RECIPIENT_SALUTATION	(373, 30)
public	string	RECIPIENT_FOR_ATTENTION	(403, 35)

4. Statement Class



public	string	RECIPIENT_DESIGNATION	(438, 35)
public	string	FILLER	(473, 190)

5.1.2 Class - FastRecord150

using Fast Record RQ 01191C (Record ID 150)

Fast Name - Statement Composite MASS

5.1.3 Class – FastRecord190

using Fast Record ID – RQ00838C

Fast Name – CTV Composite