CDRCashflow

Updated: 24 May 2016

Use the table-valued function CDRCashflow to return a cash-flow schedule for a loan with a fixed periodic payment with Conditional Prepayment Rates (CPR) and Constant Default Rates (CDR) applied.

Syntax

```vbnet
'METHOD: CDRCashflow  (1/3)
Public Shared Function CDRCashflow(
    ByVal PrinAmt As Double,
    ByVal InterestRate As Double,
    ByVal NumPmts As Integer,
    ByVal LastPmtNum As Integer,
    ByVal PmtPerYr As Integer,
    ByVal LSRates As System.Array,
    ByVal CPRRates As System.Array,
    ByVal CDRRates As System.Array,
    ByVal InterestOnly As Boolean,
    ByVal PrinPaymentMultiple As Integer,
    ByVal FirstPrinPayNo As Integer,
    ByVal PmtPayPct As Double,)


'METHOD: CDRCashflow  (2/3)
Public Shared Function CDRCashflow(
    ByVal PrinAmt As Double,
    ByVal InterestRate As Double,
    ByVal NumPmts As Integer,
    ByVal LastPmtNum As Integer,
    ByVal PmtPerYr As Integer,
    ByVal LSRates As System.Data.DataTable,
    ByVal CPRRates As System.Data.DataTable,
    ByVal CDRRates As System.Data.DataTable,
    ByVal InterestOnly As Boolean,
    ByVal PrinPaymentMultiple As Integer,
    ByVal FirstPrinPayNo As Integer,
    ByVal PmtPayPct As Double,)


'METHOD: CDRCashflow  (3/3)
Public Shared Function CDRCashflow(
    ByVal PrinAmt As Double,
    ByVal InterestRate As Double,
    ByVal NumPmts As Integer,
    ByVal LastPmtNum As Integer,
    ByVal PmtPerYr As Integer,
    ByVal LSRates_per As IList(Of Integer),
    ByVal LSRates_SMM As IList(Of Double),
    ByVal CPRRates_per As IList(Of Integer),
    ByVal CPRRates_SMM As IList(Of Double),
    ByVal CDRRates_per As IList(Of Integer),
    ByVal CDRRates_SMM As IList(Of Double),
    ByVal InterestOnly As Boolean,
    ByVal PrinPaymentMultiple As Integer,
```

```vbnet
        ByVal FirstPrinPayNo As Integer,
        ByVal PmtPayPct As Double,)
```

Arguments

PrinAmt

> the principal amount to be amortized. *PrinAmt* is an expression that returns a **Double**, or of a type that can be implicitly converted to **Double**.

InterestRate

> the annual rate of interest used to calculate the periodic payment. *InterestRate* is an expression that returns a **Double**, or of a type that can be implicitly converted to **Double**.

NumPmts

> the number of periods to be used in the calculation of the periodic payment. *NumPmts* is an expression that returns a **Integer**, or of a type that can be implicitly converted to **Integer**.

LastPmtNum

> the number of the last payment. Use @LastPmtNum for case where the number of payments for the annuity calculation is different than the actual number of payments, For example, an annuity based on 300 monthly payment which will be paid off at the end of 120 months. *LastPmtNum* is an expression that returns a **Integer**, or of a type that can be implicitly converted to **Integer**.

PmtPerYr

> the number of payments per year. *PmtPerYr* is an expression that returns a **Integer**, or of a type that can be implicitly converted to **Integer**.

LSRates

> the months and loss severity rates to be used in the calculation of the loss severity amounts. LSRates contains 2 data columns, month and rate, where 1% = .01. *LSRates* is an expression that returns a **2-dimensional array of Object** (col,row) or a **System.Data.DataTable** where the first column contains **Integer** values, or values of types that can be implicitly converted to **Integer**, and the second column contains **Double** values, or values of types that can be implicitly converted to **Double**.

LSRates_per

> the months to be used in the calculation of the loss severity amounts. *LSRates_per* is an expression of a type that implements **IList(of Integer)** including system.array, arraylist, and list.

LSRates_SMM

> the loss severity rates to be used in the calculation of the loss severity amounts, where 1% = .01. *LSRates_SMM* is an expression of a type that implements **IList(of Double)** including system.array, arraylist, and list.

CPRRates

> The months and prepayment rates to be used in the calculation of the principal prepayments. CPRRates contains 2 data columns, month and rate, where 1% = .01. *CPRRates* is an expression that returns a **2-dimensional array of Object** (col,row) or a **System.Data.DataTable** where the first column contains **Integer** values, or values of types that can be implicitly converted to **Integer**, and the second column contains **Double** values, or values of types that can be implicitly converted to **Double**.

CPRRates_per

> the months to be used in the calculation of the principal prepayments. *CPRRates_per* is an expression of a type that implements **IList(of Integer)** including system.array, arraylist, and list.

CPRRates_SMM

> the principal prepayments to be used in the calculation of principal prepayments, where 1% = .01. *CPRRates_SMM* is an expression of a type that implements **IList(of Double)** including system.array, arraylist, and list.

CDRRates

> The months and default rates to be used in the calculation of the default amounts. CDRRates contains 2 data columns, month and rate, where 1% = .01. *CDRRates* is an expression that returns a **2-dimensional array of Object** (col,row) or a **System.Data.DataTable** where the first column contains **Integer** values, or values of types that can be implicitly converted to **Integer**, and the second column contains **Double** values, or values of types that can be implicitly converted to **Double**.

CDRRates_per

> the months to be used in the calculation of the default amounts. *CDRRates_per* is an expression of a type that implements **IList(of Integer)** including system.array, arraylist, and list.

CDRRates_SMM

> the default rates to be used in the calculation of the default amounts, where 1% = .01. *CDRRates_SMM* is an expression of a type that implements **IList(of Double)** including system.array, arraylist, and list.

InterestOnly

> a **boolean** value, which when true, identifies that the principal amount is scheduled to be repaid at the end of the loan.

PrinPaymentMultiple

> the ratio of the frequency of the interest payments to the frequency of the interest payments. For example, a loan with monthly payments of interest and quarterly payments of principal would have a PrinPaymentMultiple of 3. *PrinPaymentMultiple* is an expression of type **Integer** or of a type that can be implicitly converted to **Integer**.

FirstPrinPayNo

> the payment number of the first principal payment. *FirstPrinPayNo* is an expression of type **Integer** or of a type that can be implicitly converted to **Integer**.

PmtPayPct

> a fixed percentage which is applied to the projected principal balance to calculate the projected principal payment. *PmtPayPct* is of a type **Double** or of a type that can be implicitly converted to **Double**.

Return Type
FinancialTypes.**CDRCashflow_table**

```
Class CDRCashflow_table
    Inherits Data.DataTable
    Property Item(RowIndex As Integer) As FinancialTypes.CDRSched

Class CDRSched
    Public num_pmt As Integer
    Public CPR As Double
    Public CDR As Double
    Public LS As Double
    Public cont_prin_begin As Double
    Public cont_pmt As Double
    Public cont_int_pay As Double
    Public cont_prin_pay As Double
    Public cont_prin_end As Double
    Public proj_prin_begin As Double
    Public proj_pmt As Double
    Public proj_int_pay As Double
    Public proj_prin_pay As Double
    Public prin_prepay As Double
    Public amt_default As Double
    Public amt_loss_severity As Double
    Public proj_prin_end As Double
    Public proj_amt_cashflow As Double
End Class
```

Column	Description
num_pmt	payment number
CPR	the Conditional Prepayment Rate
CDR	the Constant Default Rate
LS	the Loss Severity Rate
cont_prin_begin	When **num_pmt** = 1 then *PrinAmt* else **cont_prin_end** from the previous row
cont_pmt	the contractual periodic payment.
cont_int_pay	the contractual interest payment; *InterestRate* / *PmtPerYr* *

	cont_prin_begin
cont_prin_pay	the contractual principal payment.
proj_prin_begin	When **num_pmt** = 1 then *PrinAmt* else **proj_prin_end** from the previous row
proj_pmt	the projected periodic payment. The **proj_pmt** is recalculated every period to reflect the adjustment to the principal balance arising from principal prepayments and defaults: PMT*(InterestRate / PmtPerYear, NumPmts* - **num_pmt** + 1, -**proj_prin_begin**, 0, 0).
proj_int_pay	the projected interest payment; *InterestRate / PmtPerYr* * **proj_prin_begin**
proj_prin_pay	the projected principal payment.
prin_prepay	(1- POWER(1-**CPR**,1 / *PmtPerYr*) * (**proj_prin_begin** – **proj_prin_pay**)
amt_default	(**proj_prin_begin** – **proj_prin_pay**) * (1 - POWER(1 - **CDR**, 1 / *PmtPerYr*))
amt_loss_severity	**amt_default** * *LossSeverity*
proj_prin_end	the projected principal ending balance; **proj_prin_begin** – **proj_prin_pay** - **prin_prepay** - **amt_default**
proj_amt_cashflow	the projected cash flow amount; **proj_pmt** + **prin_prepay** + **amt_default** - **amt_loss_severity**

Remarks

- If FirstPrinPayNo is NULL then FirstPrinPayNo = 1.
- If PrinAmt is NULL then PrinAmt = 0.
- If InterestRate is NULL then InterestRate = 0.
- If NumPmts is NULL then NumPmts = 0.
- If LastPmtNum is NULL then LastPmtNum = NumPmts.
- If InterestOnly is NULL then InterestOnly = FALSE.
- If PrinPaymentMultiple is NULL then PrinPaymentMultiple = 1.
- If FirstPrinPayNo is NULL then FirstPrinPayNo = PrinPaymentMultiple.
- If NumPmts < 1 then no rows are returned.
- If PrinPaymentMultiple < 1 then no rows are returned.
- If FirstPrinPayNo < 1 then no rows are returned.
- PmtPerYr must be 1, 2, 3, 4, 6, or 12.
- If LSRates returns NULL or no rows then LS is set to zero.
- If CDRRates returns NULL or no rows then CDR is set to zero.
- If CPRRates returns NULL or no rows then CPR is set to zero.

Examples

Find examples that illustrate how to call this function in the demo application bundled with the XLeratorDLL trial download.

See Also

- AMORTRATE - Constant daily effective rate for bond/loan amortization
- AMORTSCHED - Generate amortization schedule of a loan
- Balloon - Schedule with periodic interest payments and principal repaid at maturity
- Bullet - Schedule with single interest and principal payment at maturity
- ConstantCashFlow - Schedule with equal periodic cash flows
- ConstantCashFlowFR - Schedule for a loan with a fixed maturity date and annuity-style payments
- ConstantPaymentAmount -Schedule with no maturity with fixed periodic payment amount
- ConstantPrincipal - Schedule with fixed maturity date where the periodic principal payment is calculated on a straight-line basis
- ConstantPrincipalAmount - Schedule with no fixed maturity with a fixed periodic principal payment
- ConstantPrincipalRate - schedule with no fixed maturity where a fixed percentage principal payment
- CONSTPRINAMORT - Schedule of a loan with a fixed principal repayment
- NPD - Next payment date of a loan
- NPNO - Next payment number of a loan
- PAYMENTPERIODS - Number of months until first payment date, start of grace period, end of grace period, and total number payments for a loan
- PERIODRATE - Adjust the nominal rate of a loan
- PPD - Previous payment date of a loan
- PPNO - Previous payment number of a loan
- UNEQUALLOANPAYMENTS - Schedule for a loan where interest and principal payment frequencies differ.