Scanduc|25

Maintenance Tips & Tricks

John Hartley

Who is John Hartley?

- Dataflex developer since 1999
- C# .NET developer since 2014
- Freedom IT founded 2014
- FrontIoT sub-contractor since 2022

Tips & Tricks - System Maintenance

- A mandatory part of a developer’s job description
- Less interesting than developing new features
- It can (eventually) be fun and easy
- Reducing maintenance is exponential (and vice-versa)

Minimising Maintenance

- Knowledge - experience
- Knowledge - high-level documentation
- Instinct - consistency
- Readability - low-level documentation
- Architecture/Technique

Part 1 - Basic Tips

Tip 1 - Constants

Don’t do this:

```
    Procedure Fill_List
        Forward Send Fill_List
        Send Add_Table_Value  "NONE"    "None established"
        Send Add_Table_Value  "COD"     "COD"
        Send Add_Table_Value  "NET30"   "Net 30"
        Send Add_Table_Value  "NET60"   "Net 60"
        Send Add_Table_Value  "NET90"   "Net 90"
        Send Add_Table_Value  "PREPAY" "Pre-payment required"
    End_Procedure
```

Do this instead:

```
Define C_ORDER_TERM_NONE   for "NONE"
Define C_ORDER_TERM_COD    for "COD"
Define C_ORDER_TERM_NET30  for "NET30"
Define C_ORDER_TERM_NET60  for "NET60"
Define C_ORDER_TERM_NET90  for "NET90"
Define C_ORDER_TERM_PREPAY for "PREPAY"
—-----------------------------------
    Procedure Fill_List
        Forward Send Fill_List
        Send Add_Table_Value  C_ORDER_TERM_NONE   "None established"
        Send Add_Table_Value  C_ORDER_TERM_COD    "COD"
        Send Add_Table_Value  C_ORDER_TERM_NET30  "Net 30"
        Send Add_Table_Value  C_ORDER_TERM_NET60  "Net 60"
```

Tip 2 - Numeric Status/Checkbox Fields

Don’t do this:

```
Define C_ORDER_TERM_NONE   for "NONE"
Define C_ORDER_TERM_COD    for "COD"
Define C_ORDER_TERM_NET30  for "NET30"
Define C_ORDER_TERM_NET60  for "NET60"
Define C_ORDER_TERM_NET90  for "NET90"
Define C_ORDER_TERM_PREPAY for "PREPAY"
```

Do this instead:

```
Enum_List
    Define C_ORDER_TERM_NONE
    Define C_ORDER_TERM_COD
    Define C_ORDER_TERM_NET30
    Define C_ORDER_TERM_NET60
    Define C_ORDER_TERM_NET90
    Define C_ORDER_TERM_PREPAY
    Define C_ORDER_TERM_END
End_Enum_List
```

Tip 3 - Create Separate Files

Don’t do this:

cOrderHeaderDataDictionary.dd

```
Enum_List
        Define C_ORDER_TERM_NONE
        Define C_ORDER_TERM_COD
        –----------------------
End_Enum_List

Object Terms_table is a DescriptionValidationTable

        Procedure Fill_List
        Forward Send Fill_List
        Send Add_Table_Value C_ORDER_TERM_NONE   "None established"
        Send Add_Table_Value C_ORDER_TERM_COD    "COD"
        –-----------------------------------------
    End_Procedure

End_Object
```

Tip 3 - Create Separate Files

Do this instead:

cOrderHeaderDataDictionary.dd

```
Use ValidationTable\OrderDescriptionValidationTableOrderTerm.pkg
```

OrderDescriptionValidationTableOrderTerm.pkg

```
Use ValidationTable\cOrderDescriptionValidationTableOrderTerm.pkg

Object oOrderDVTOrderTerm is a cOrderDVTOrderTerm
End_Object
```

cOrderDescriptionValidationTableOrderTerm.pkg

```
Use Ddvaltbl.pkg
Use DataHolder\OrderEnumerationOrderTerm.pkg

Class cOrderDVTOrderTerm is a DescriptionValidationTable

    Procedure Fill_List
        Forward Send Fill_List
        Send Add_Table_Value C_ORDER_TERM_NONE "None established"
```

Tip 4 - Comment Files into ‘Regions’

Don’t do this:

```
Use Ddvaltbl.pkg
Use DataHolder\OrderEnumerationOrderTerm.pkg

Class cOrderDVTOrderTerm is a DescriptionValidationTable

    Procedure Fill_List
        Forward Send Fill_List
        Send Add_Table_Value C_ORDER_TERM_NONE   "None established"
        Send Add_Table_Value C_ORDER_TERM_COD    "COD"
```

Do this instead:

```
// DF
Use Ddvaltbl.pkg

// Order
Use DataHolder\OrderEnumerationOrderTerm.pkg

Class cOrderDVTOrderTerm is a DescriptionValidationTable

        // ValidationTable
        Procedure Fill_List
        Forward Send Fill_List
```

Part 2 - Advanced Tips

The OO Maintenance Paradox

Adhering to the well established, object oriented principles makes systems more maintainable

Tip 5 - Create Abstractions

Don’t do this:

```
// DF
Use Ddvaltbl.pkg

// Order
Use DataHolder\OrderEnumerationOrderTerm.pkg

Class cOrderDVTOrderTerm is a DescriptionValidationTable

        // ValidationTable
        Procedure Fill_List
        Forward Send Fill_List
```

Do this instead:

```
// Global
Use MixinDescriptionValidationTableEnumeration.pkg

// Order
Use DataHolder\OrderEnumerationOrderTerm.pkg

Class cOrderDVTOrderTerm is a DescriptionValidationTable

    // Global
    Import_Class_Protocol cMixinDVTEnumeration
```

Tip 6 - Use Mixins for Abstractions

Don’t do this:

cDescriptionValidationTableEnumerationBase.pkg

```
// DF...
Use DDvalTbl.pkg

{ClassType = Abstract}
Class cDVTEnumerationBase is a DescriptionValidationTable
—------------------------------------------------
End_Class
```

cDescriptionValidationTableWithDescriptionPrefixedWithCode.pkg

```
// DF...
Use DDvalTbl.pkg

Class cDVTWithDescrPrefixedWithCode is a DescriptionValidationTable
—------------------------------------------------------------
End_Class
```

Tip 6 - Use Mixins for Abstractions

Don’t even do this:

cDescriptionValidationTableGOD.pkg

```
// DF...
Use DDvalTbl.pkg

Class cDVTGod is a DescriptionValidationTable

        Procedure Construct_Object
        Forward Send Construct_Object

        Property Boolean pbDVTGodUsesEnumeration
        Property Boolean pbDVTGodHasDescriptionPrefixedWithCode
        Property Boolean pbDVTGodMickeyMouseState
        Property Boolean pbDVTGodPokeInTheEyeState
        Property Boolean pbDVTGodSomeOtherState
        Property Boolean pbDVTGodYetAnotherState

        //TODO: add more states here as God becomes all powerful
    End_Procedure

    //-----------------------------------------------------

End_Class
```

Tip 6 - Use Mixins for Abstractions

Do this instead:

cMixinDescriptionValidationTableEnumeration.pkg

```
// DF...
Use DDvalTbl.pkg

Class cMixinDVTEnumeration is a Mixin
—————————————————————————
End_Class
```

cMixinDescriptionValidationTableWithDescriptionPrefixedWithCode.pkg

```
// DF...
Use DDvalTbl.pkg

Class cMixinDVTWithDescrPrefixedWithCode is a Mixin
—————————————————————————————————————————
End_Class
```

Tip 7 - Use Abstract Methods (events)

Don’t do this:

```
Class cMixinDVTEnumeration is a Mixin

    // Constructor
    Procedure mDefine_MixinDVTEnumeration

        Property Short piMixinDVTEnumerationLast
    End_Procedure
```

Do this instead:

```
Class cMixinDVTEnumeration is a Mixin

    // Event
    Procedure mMixinDVTEnumerationGetLast Short ByRef iLastOut

        Error 300 “‘mMixinDVTEnumerationGetLast’ not implemented!”
    End_Procedure
```

Tip 8 - Split Jobs into Separate Classes

Don’t do this:

```
Class cOrderDVTOrderTerm is a DescriptionValidationTable

    // Global
    Import_Class_Protocol cMixinDVTEnumeration

    // cMixinDVTEnumeration event
    Procedure mMixinDVTEnumerationGetDescr UShort            iType ;
                                           String ByRef sDescrOut
        Case Begin

            Case (iType = C_ORDER_TERM_COD)

                Move "COD" to sDescrOut
                Case Break
            //TODO: add more cases here
            Case Else

                Move "None established" to sDescrOut
        Case End
    End_Procedure
```

Tip 8 - Split Jobs into Separate Classes

Do this instead:

```
Class cOrderDVTOrderTerm is a DescriptionValidationTable

    // Global
    Import_Class_Protocol cMixinDVTEnumeration

    // Constructor
    Procedure Construct_Object

        Forward Send Construct_Object

        Object oCnvt is a cOrderTermConverter
        End_Object

    End_Procedure

    // cMixinDVTEnumeration event
    Procedure mMixinDVTEnumerationGetDescr UShort           iType ;
                                           String ByRef sDescrOut

        Send mConvert of oCnvt iType (&sDescrOut)
    End_Procedure
```

Split Jobs Into Separate Classes - Why?

- Reusability
- Substitution
- Testing

Minimising Maintenance - Testing

- The compiler
- Unit tests
- Fast fail
- Human (developer)
- Human (tester)
- Human (user)

Demo Time

Scanduc 25

Thank you!

Are there any questions?