

Autor & Copyright: Dipl.-Ing. Harald Nahrstedt

Version: 2016 / 2019 / 2021 / 365

Erstellungsdatum: 09.09.2022

Überarbeitung: 01.12.2023

Beschreibung:

Die *SplitParts*-Objektliste enthält eine Sammlung von *SplitPart*-Objekten. Durch die Unterbrechungen von Vorgängen entstehen Vorgangsteile, die auch als Partionen bezeichnet werden, mit einem Anfangs- und Enddatum.

Anwendungs-Datei: 17-01-03\_SplitParts.mpp

## 1 SplitParts-Auflistung

Die *SplitParts*-Objektliste enthält eine Sammlung von *SplitPart*-Objekten. Durch die Unterbrechungen von Vorgängen entstehen Vorgangsanteile, die Project als *SplitPart*-Objekte verwaltet, mit einem Anfangs- und Enddatum (Bild 1).

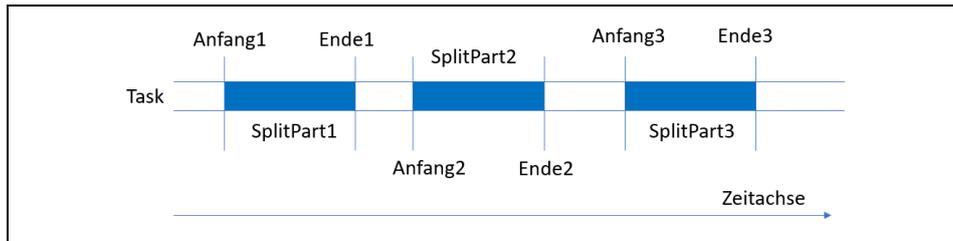


Bild 1. Aufteilung eines Vorgangs

Die *SplitParts* sind Unterobjekte des *Task*-Objekts.

Mit einer *For-Each-Next*-Schleife können die einzelnen *SplitPart*-Objekte der *Task*-Objekte angesprochen werden.

Codeliste 1. Die Prozedur nennt die Anzahl von *SplitParts* für jeden Vorgang im Projekts

```
Sub NumberOfSplitParts()  
    Dim objTask As Task  
  
    For Each objTask In ActiveProject.Tasks  
        If Not (objTask Is Nothing) Then  
            MsgBox objTask.Name & ": " & _  
                objTask.SplitParts.Count  
        End If  
    Next  
End Sub
```

Das Ergebnis wird im Direktfenster ausgegeben (Bild 2).

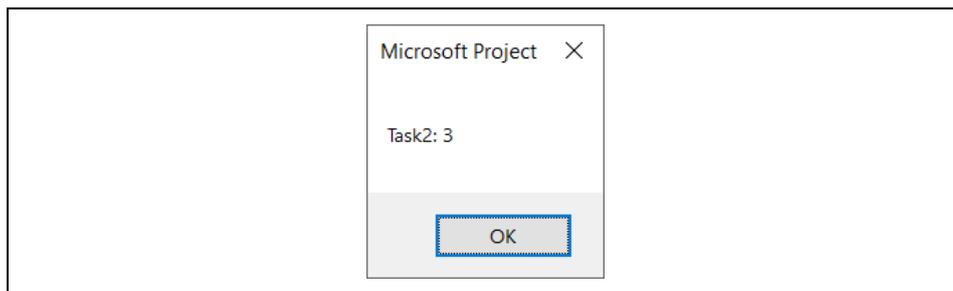


Bild 2. Beispiel einer Ausgabe

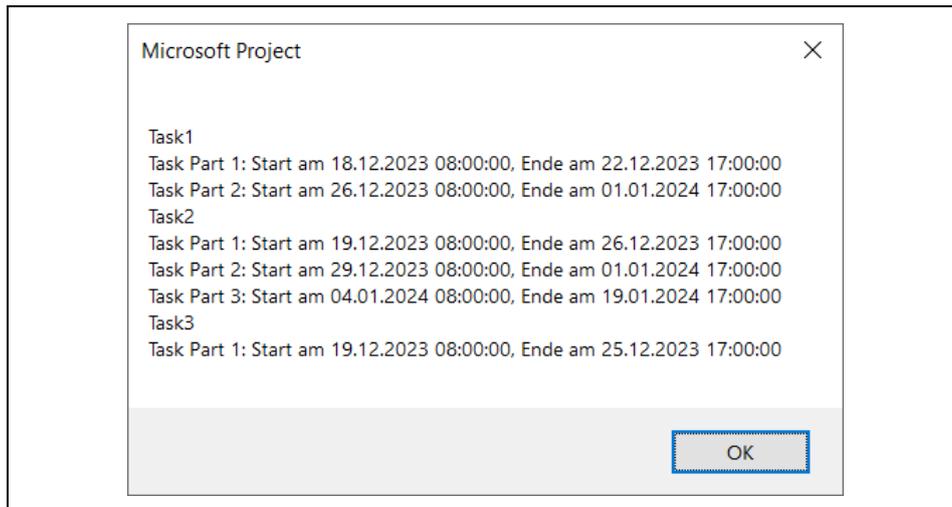
## 2 SplitPart-Eigenschaften

Neben der *Count*-Eigenschaft liefert die *Application*-Eigenschaft das dazugehörige Objekt. Mit der *Item*-Eigenschaft wird ein *SplitPart* aus der Objektliste ausgewählt. Die Eigenschaften *Start* und *Finish* liefern Anfangs- und Enddatum des *SplitParts*.

*Codeliste 2. Die Prozedur zeigt alle SplitParts eines Projekts*

```
Sub ViewAllSplitParts()  
    Dim objTask As Task  
    Dim lPart As Long  
    Dim sParts As String  
  
    sParts = ""  
    For Each objTask In ActiveProject.Tasks  
        If Not (objTask Is Nothing) Then  
            sParts = sParts & objTask.Name & vbCrLf  
            For lPart = 1 To objTask.SplitParts.Count  
                With objTask  
                    sParts = sParts & "Task Part " & lPart & _  
                        ": Start am " & _  
                        .SplitParts(lPart).Start & ", Ende am " & _  
                        .SplitParts(lPart).Finish & vbCrLf  
                End With  
            Next lPart  
        End If  
    Next  
    MsgBox sParts  
End Sub
```

Die Prozedur sammelt alle *SplitParts*, um sie am Ende der Prozedur anzuzeigen (Bild 3).



*Bild 3. Ausgabebeispiel*

Die *Parent*-Eigenschaft liefert das Vorgangsobjekt, zu dem das *SplitPart*-Objekt gehört.

### 3 SplitPart-Methoden

Die einzig wichtige Methode des *SplitPart*-Objekts ist die *Add*-Methode. Ihre Syntax lautet

```
SplitPart.Add (StartSplitPartOn, EndSplitPartOn)
```

Tabelle 1. Die erforderlichen Parameter der *Add*-Methode

Parameter	Datentyp	Description
StartSplitPartOn	Variant	Das Startdatum des Vorgangsteils.
EndSplitPartOn	Variant	Das Enddatum des Aufgabenteils. Wenn <i>EndSplitPartOn</i> am oder vor dem mit <i>StartSplitPartOn</i> angegebenen Datum liegt, wird der Teil nicht erstellt.

Im Beispiel soll der Vorgang *Task3* gesplittet werden (Bild 4).

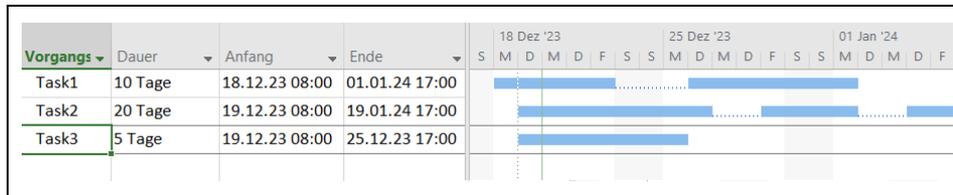


Bild 4. *Task3* vor der Splittung

Die folgende Prozedur splittet den Vorgang *Task3* mit den angegebenen Daten. Dazu muss vorher *Task3* ausgewählt werden.

Codeliste 3. Die Prozedur splittet den Vorgang *Task3*

```
Sub SplitTask3()
    ActiveCell.Task.Split "21.12.2023", "02.01.2024"
End Sub
```

Nach dem Aufruf von *SplitTask3* ist *Task3* entsprechend geteilt (Bild 5).

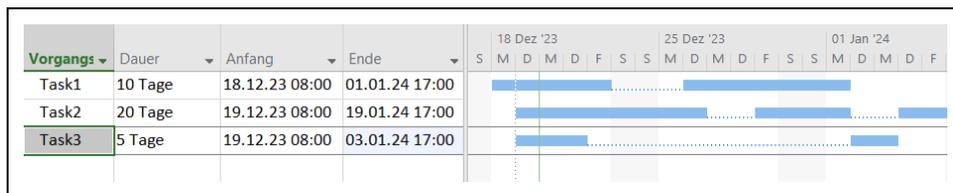


Bild 5. *Task3* gesplittet