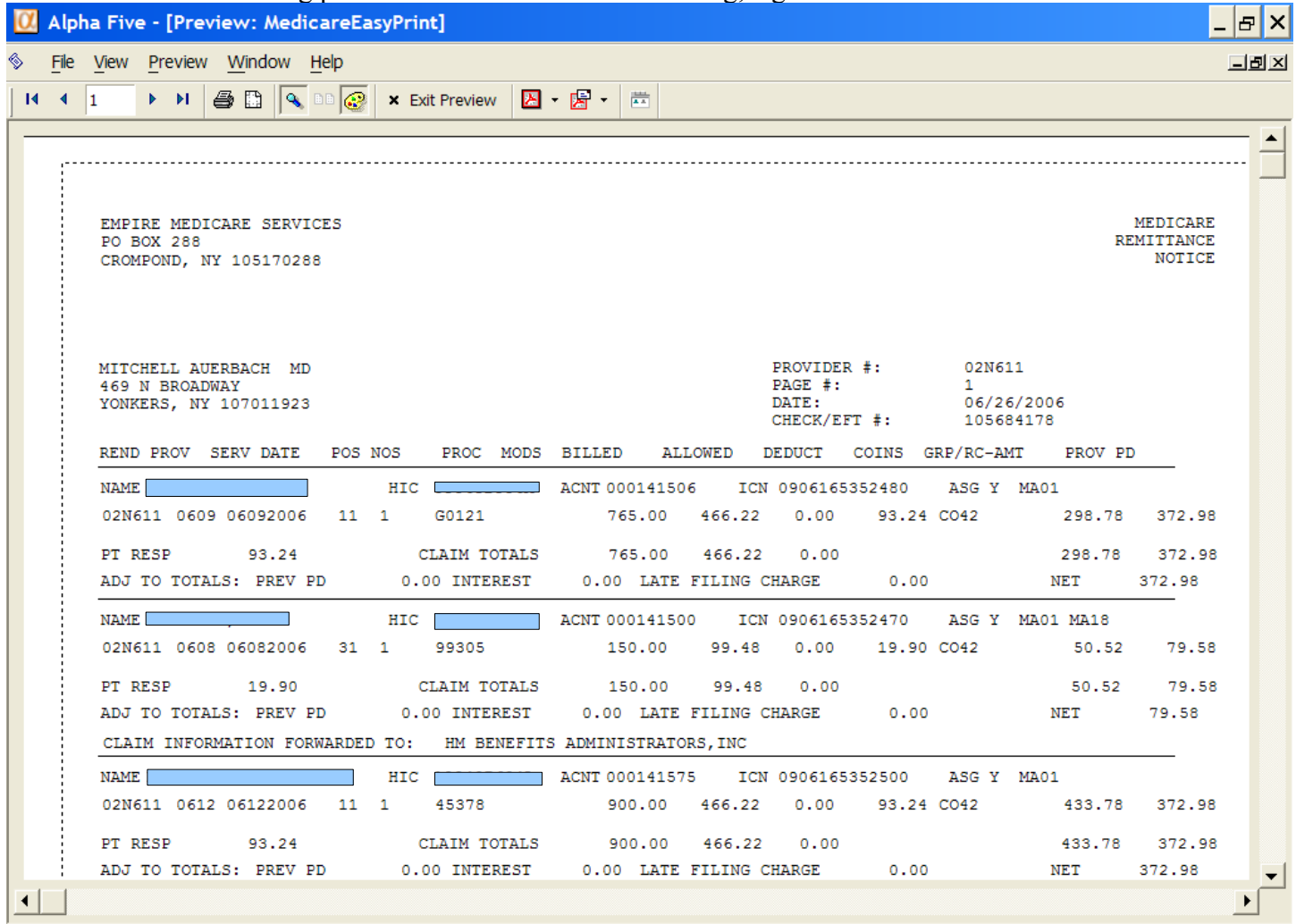


Scope of the problem:

The counterpart to electronic billing for medical offices is electronic remittance notices. The electronic remittance file lists each claim, tells what is not being paid, what is, and how much, and why.

Each individual claim remittance is usually accompanied by one to several “remarks” and “reasons,” such as “Duplicate claim/service” or “The diagnosis is inconsistent with the patient's gender.” All in all, there are over 900 possible remarks and reasons defined for remittances. Rather than provide the full text of the remark, the insurer only sends predefined *remark codes* or *reason codes* in the electronic file.

The standard *printed* remittance advice for multiple claims has included the remark codes with an explanation of the remarks used at the end of the (paper) remittance advice. My goal was to create an Alpha Five report that mimics the one that billing personnel are accustomed to reading, e.g.:



In this image, I've blocked out patient names and insurance identification numbers. This Alpha Five report looks just like the notice we get from Medicare, down to the font selection. The remarks seen here include MA01, MA18, and CO42. The last page of the report shows the explanation of the remarks:

Alpha Five - [Preview: MedicareEasyPrint]

File View Preview Window Help

5 Exit Preview

NAME	HIC	ACNT	ICN	ASG	Y	MA01	MA27
92D591 0610 06102006	21 1 99231	000141550	0906165352560	7.74	CO42	26.30	30.96
PT RESP	7.74	CLAIM TOTALS	65.00 38.70 0.00			26.30	30.96
ADJ TO TOTALS: PREV PD	0.00 INTEREST	0.00 LATE FILING CHARGE	0.00			NET	30.96

NAME	HIC	ACNT	ICN	ASG	Y	MA01	
92D591 0609 06092006	24 1 45385	000141523	0906165352520	67.73	CO58	1146.33 270.94	
92D591 0609 06092006	24 1 45381 59 51	1000.00	28.66 0.00	5.73		971.34 22.93	
PT RESP	73.46	CLAIM TOTALS	2485.00 367.33 0.00			2117.67 293.87	
ADJ TO TOTALS: PREV PD	0.00 INTEREST	0.00 LATE FILING CHARGE	0.00			NET	293.87

NAME	HIC	ACNT	ICN	ASG	Y	MA130	MA15
92D591 0622 06222006	11 0 82272	000141931	0906177600362	0.00	CO16	10.00	0.00
PT RESP	0.00	CLAIM TOTALS	10.00 0.00 0.00			10.00	0.00
ADJ TO TOTALS: PREV PD	0.00 INTEREST	0.00 LATE FILING CHARGE	0.00			NET	

MA01 : If you do not agree with what we approved for these services, you may appeal our decision. To make sure that we are fair to you, we require another individual that did not process your initial claim to conduct the appeal. However, in order to be eligible for an appeal, you must write to us within 120 days of the date you received this notice, unless you have a good reason for being late.  
 42 : Charges exceed our fee schedule or maximum allowable amount.  
 MA18 : The claim information is also being forwarded to the patient's supplemental insurer. Send any questions regarding supplemental benefits to them.  
 18 : Duplicate claim/service.  
 MA72 : The patient overpaid you for these assigned services. You must issue the patient a refund within 30 days for the difference between his/her payment to you and the total of the amount shown as patient responsibility and as paid to the patient on this notice.  
 50 : These are non-covered services because this is not deemed a 'medical necessity' by the payer.  
 MA27 : Missing/incomplete/invalid entitlement number or name shown on the claim.  
 49 : These are non-covered services because this is a routine exam or screening procedure done in conjunction

The electronic remittance advice is known as an ANSI 835 file, defined in a 252-page document that makes great bedtime reading. I won't go into the parsing of the file; suffice it to say that the file is read and is parsed into an Alpha Five table, imaginatively called "ansi835.dbf", whose structure includes a 30-character field called "reason\_codes" and another called "remark\_codes". The fields get filled with whatever remark or reason codes accompany the bill's remittance advice, as in this example from a different insurer:

Cpt	Mod1	Mod2	Svc_date	Reason_codes	Line_charge	Payment	Adjustment	Deductible	Pat_res
45385			01/17/2006	CO30	1485.00	0.00	1485.00		
99244			01/18/2006	CO15	250.00	0.00	250.00		
99214			01/25/2006	CO15	125.00	0.00	125.00		
82270			01/25/2006	CO15 CO16	10.00	0.00	10.00		
45378			03/17/2006	CO15	900.00	0.00	900.00		
45385			03/21/2006	CO15	1485.00	0.00	1485.00		
43246			03/21/2006	CO15	600.00	0.00	600.00		
99213			03/23/2006	CO15	75.00	0.00	75.00		
36415			03/23/2006	CO15	10.00	0.00	10.00		
99212			04/06/2006	CO15	60.00	0.00	60.00		
36415			04/06/2006	CO15 COB18	15.00	0.00	15.00		
45378			04/03/2006	CO18	900.00	0.00	900.00		
99284			02/17/2005	CO15	200.00	0.00	200.00		
43235			03/23/2006	CO15	650.00	0.00	650.00		
45385			04/05/2006	CO15	1485.00	0.00	1485.00		
45380			04/05/2006	CO15	1050.00	0.00	1050.00		
J7050			04/05/2006	CO15	15.00	0.00	15.00		
99000			04/05/2006	CO15 COB18	10.00	0.00	10.00		
45385			10/18/2005	CO45	1485.00	0.00	1485.00		
99244			03/20/2006	CO45	250.00	8.10	241.90		
82270			03/20/2006	CO45	10.00	0.00	10.00		
82270			03/20/2006	CO15	10.00	0.00	10.00		

I *could* have made the “remark” field into a memo field and filled it with the expanded text of all the remarks, but that would create a very long report indeed, and one in which the essential remittance information is lost in a blizzard of redundant remarks. After all, an experienced biller soon learns that “CO18” means a “duplicate claim/service” and does not need to see it written out; the “long form” of the remarks at the end of the report is a reference for the inexperienced biller as well as for unfamiliar remark codes.

The plan, then, is to read each claim in sequence. If there is a previously unencountered remark or reason code, I should add it to the list of reason codes I need to expand at the end of the report. I could do this in Xbasic using two little-used features of Alpha Five: the report's **OnRecord** event and **collections**.

I had to remind myself what a **collection** is: it's an array whose index is character, not numeric. For example, I could set up this collection:

```
dim members as U
members.set("weiss", "daniel")
members.set("talbott", "jay")
members.set("rochford", "barry")
```

You can retrieve an element of a collection with its index:

```
? members.get("rochford")
= "barry"
```

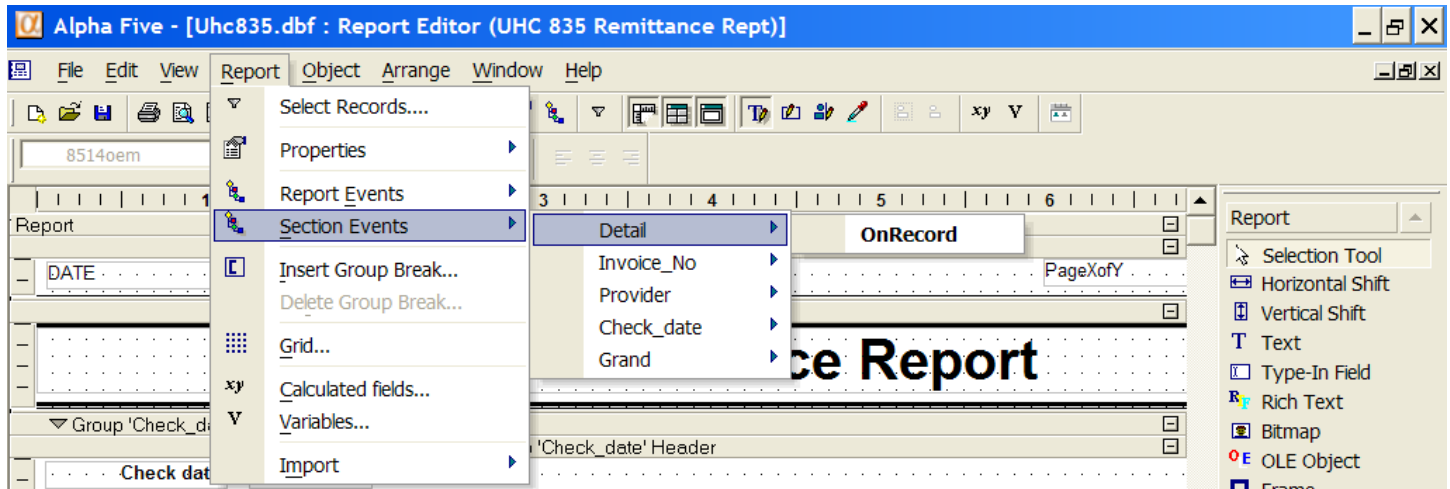
And you can see if an element exists in the collection:

```
? members.exist("rabins")
= .F.

? members.exist("talbott")
= .T.
```

So now we know everything we need to know about collections.

The event we'll write code for is the Detail Section's **OnRecord** event. The **OnRecord** event fires each time a new detail record is read:

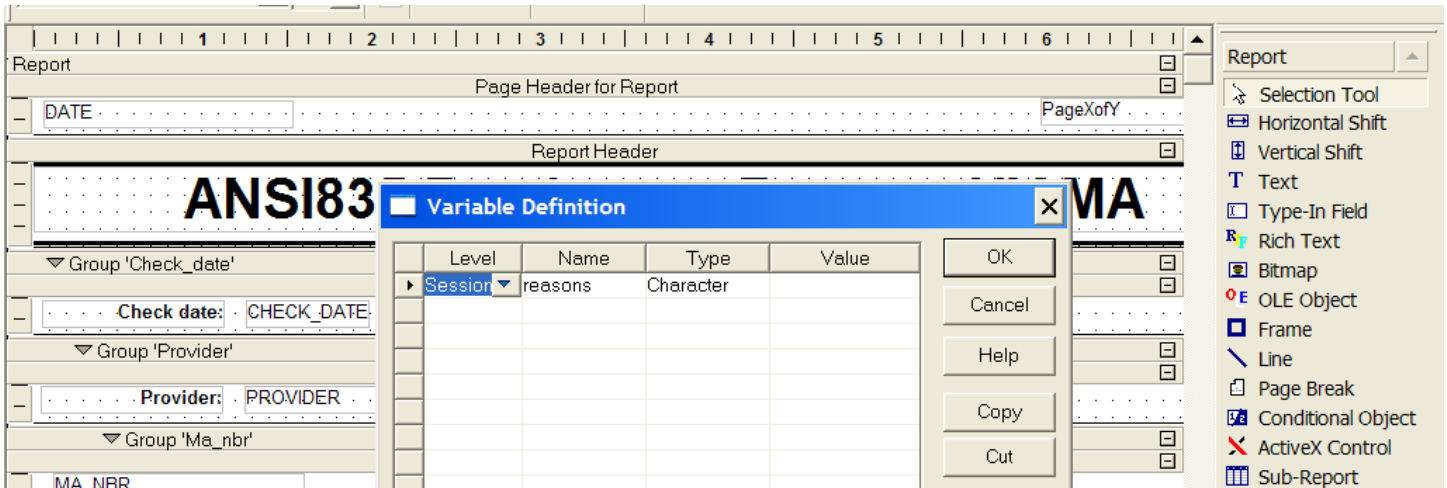


The code for one such event follows:

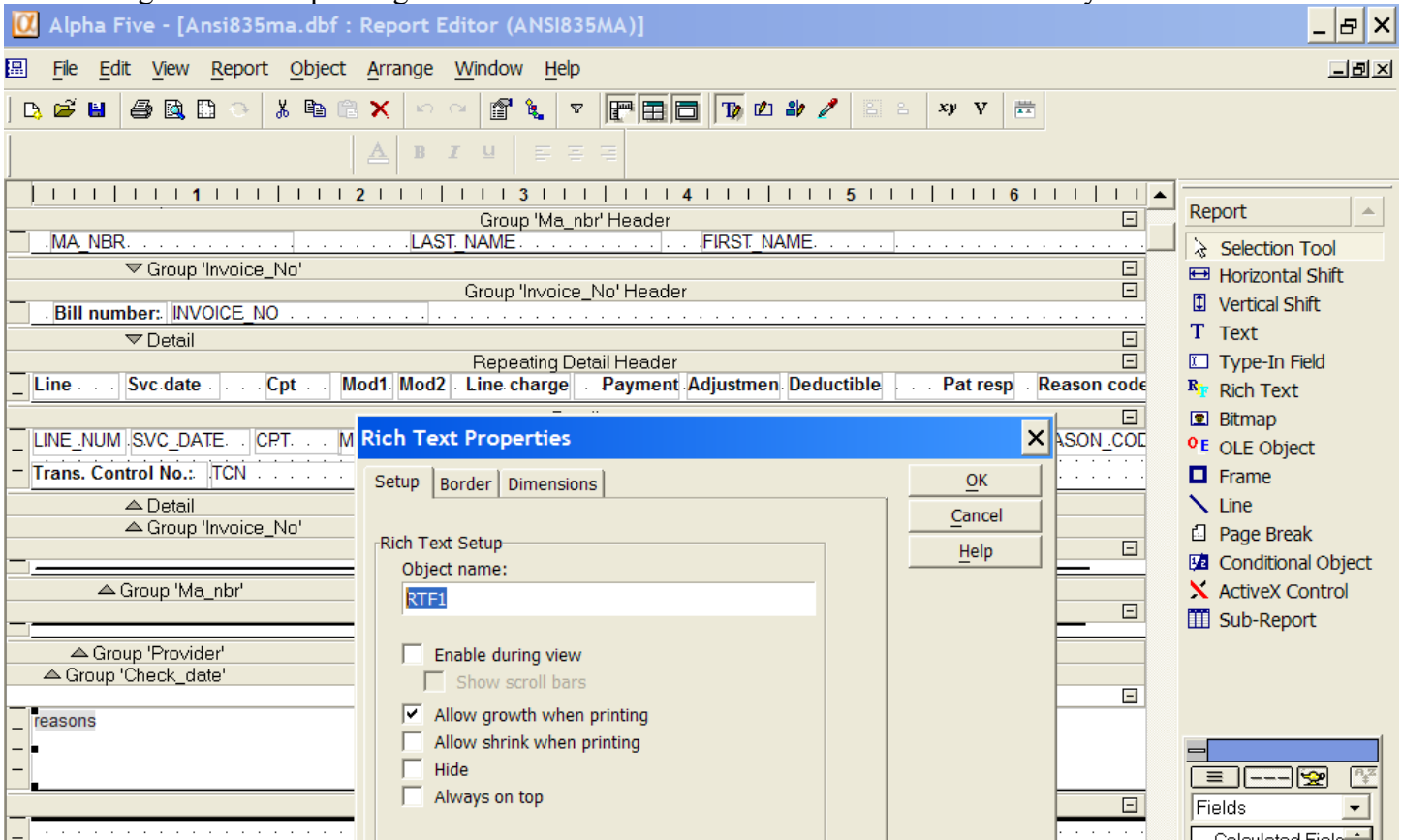
```
dim shared reasonsU as U
dim shared reasons as C
dim remark_code as c
dim reason_code as c
if ansi835ma->reason_codes>""
for i=1 to w_count(ansi835ma->reason_codes," CO")
    reason_code=trim(word(ansi835ma->reason_codes,i," CO"))
    if reason_code>"" then
        if reasonsU.get(reason_code)="" then
            reasonsU.set(reason_code,reason_code)
            reasons=reasons+reason_code+" : "+\
            table.external_record_content_get("mc_msg",\

            "word(Message,1,crlf())", "", "msg_code="+quote(reason_code)+crlf()
            end if
        end if
    end if
end if
next
end if
```

The 932 different remark codes and their explanations are in the table, "mc\_msg.dbf". Since this script is called once for each record, the collection **reasonsU** has to be defined as **shared** so it persists from one record to the next. I build a shared character string, **reasons**, that is also defined in the report's Variables as a "Session" variable:



Finally, I placed the “reasons” character variable in a rich text field in the report footer, making sure to check off “allow growth while printing” so that the rich text field could accommodate as many reasons as needed:



And that's how I learned to appreciate collections and the **OnRecord** event!