

**NAV
TECH
DAYS
2016**

mibuso.com

BAD HABITS OF NAV DEVELOPERS

ERIC WAUTERS
VJEKOSLAV BABIĆ
GARY WINTER

WHEN YOU ARE PASSIONATE ABOUT MICROSOFT DYNAMICS NAV | www.navtechdays.com

**NAV
TECH
DAYS
2016**

mibuso.com

If you're
happy
& you
know
it



clap your... oh



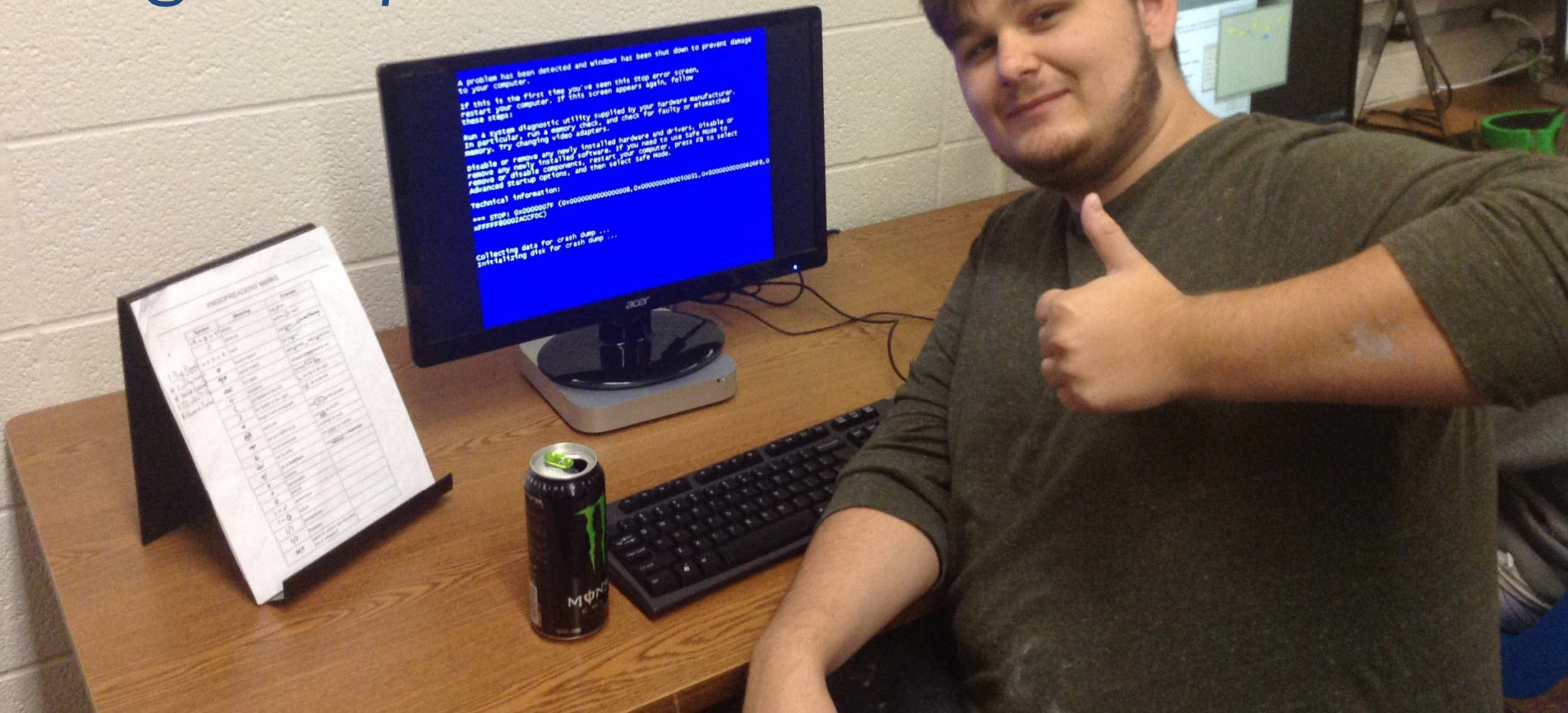
*Builders often
live in unfinished
houses*

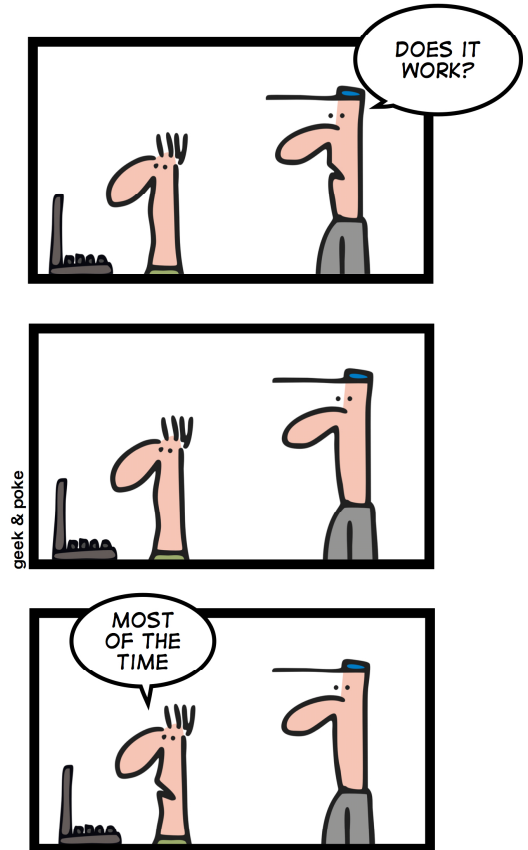
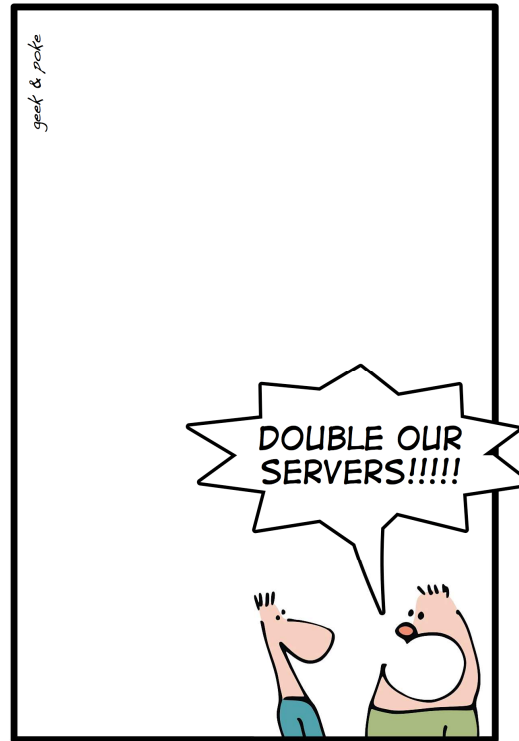
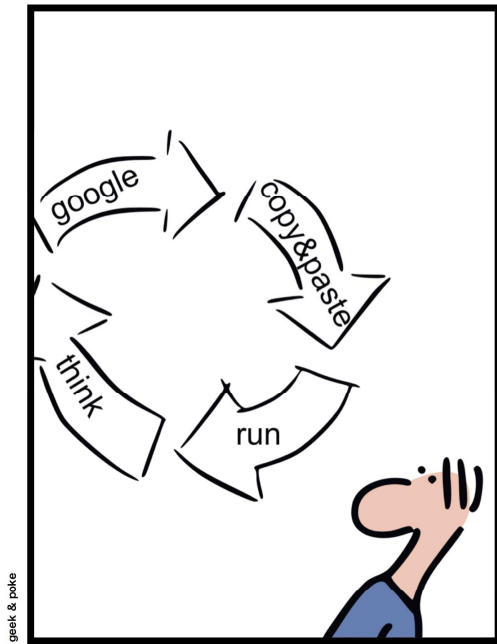
*Chefs rarely cook
extravagant meals at home*





Developers *recognize*
good practice





They just don't always get around to putting it into production environments

*Developers develop bad habits
because...*

*they practice reactive
development*



*Unwillingness to pull
on loose threads*

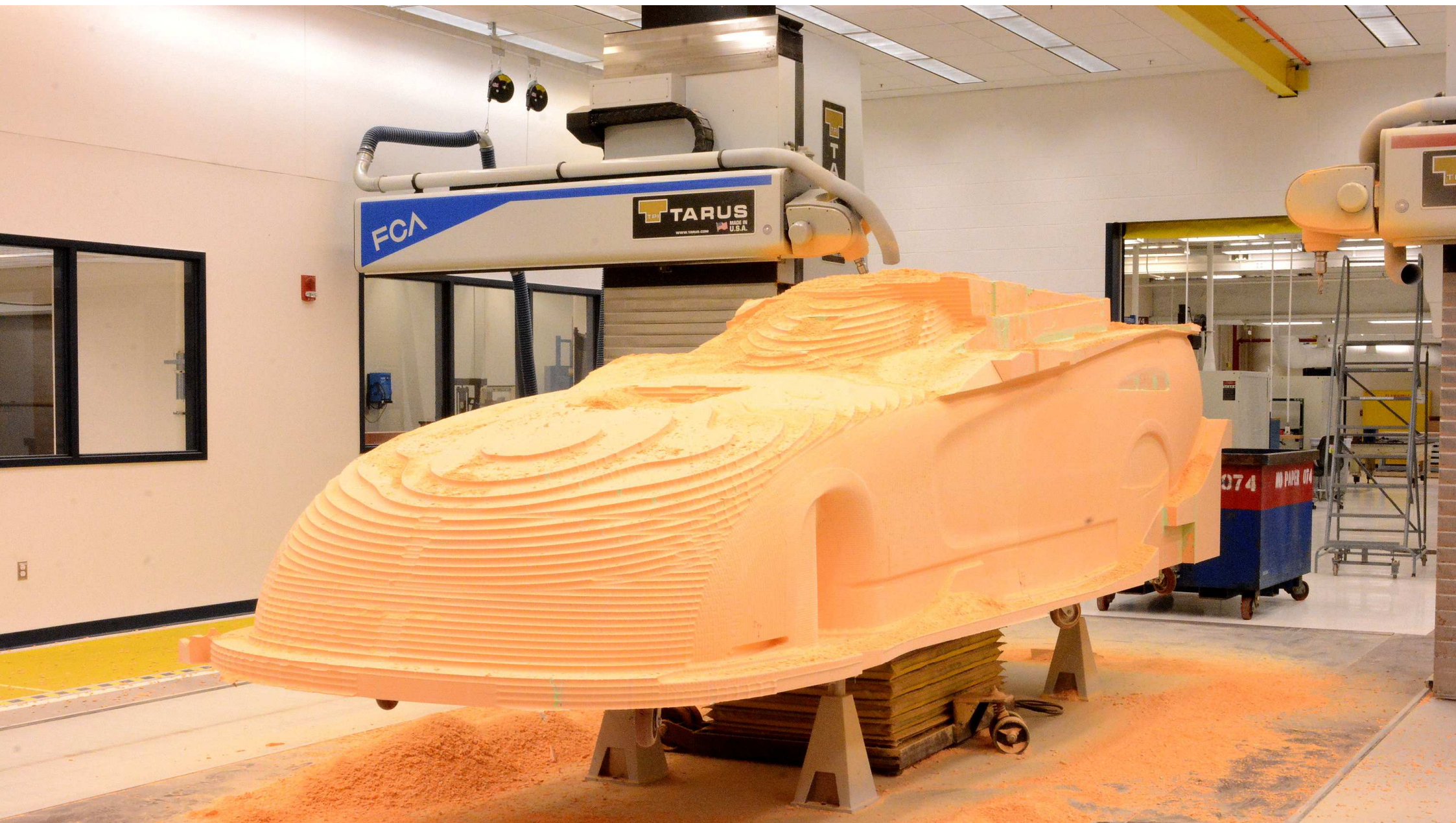




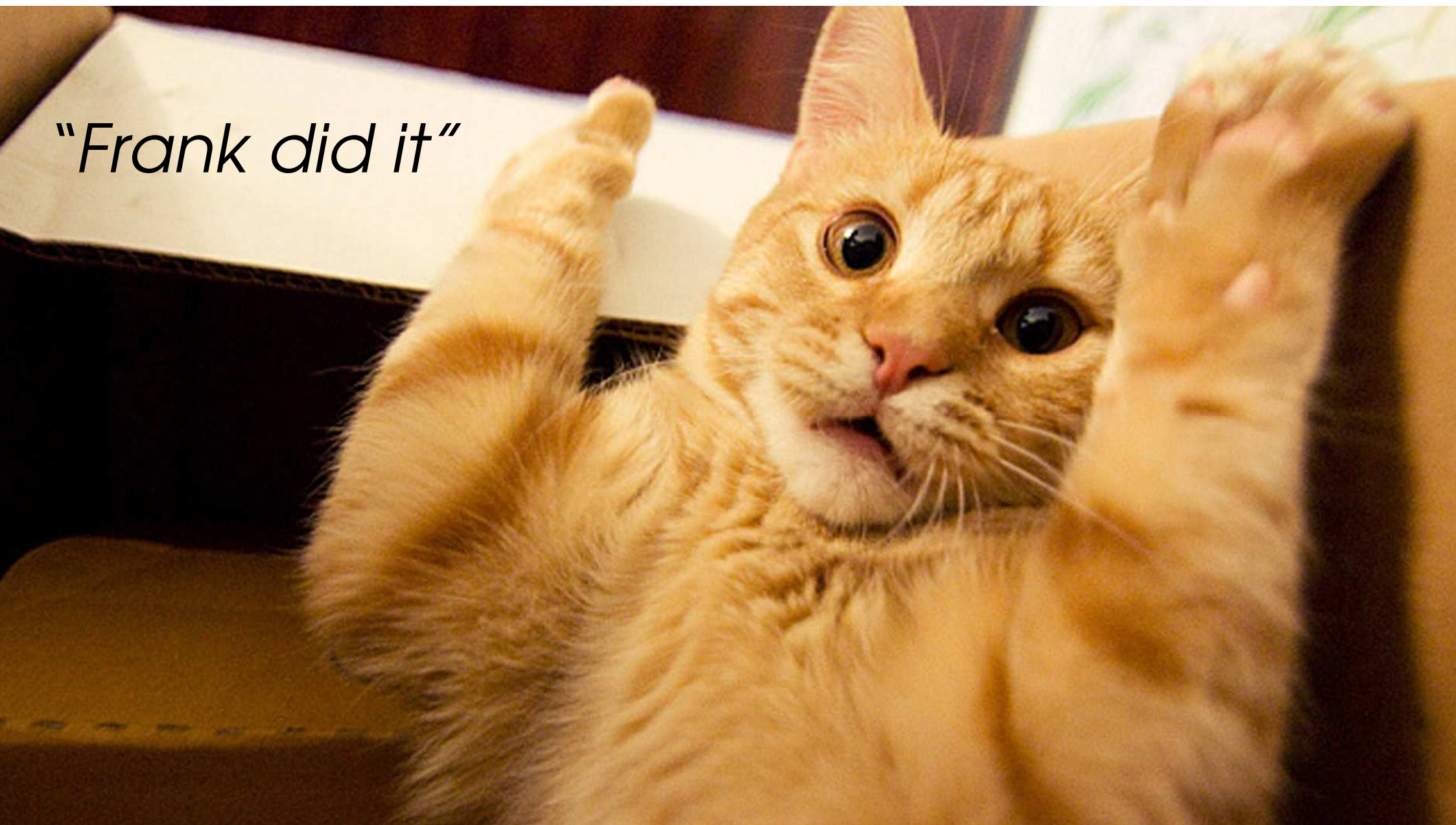
don't have time

*“This is only test code to test a prototype,
it will never end up in production”*





"Frank did it"



*It's considered a low
priority requirement*



A photograph showing a group of business professionals in a meeting. In the foreground, a man in a dark suit is leaning forward with his head down, resting on the table. Behind him, a woman in a white shirt and blue blazer is covering her face with her hands. Further back, another man is also covering his face. The scene suggests a meeting where the participants are frustrated or exhausted, likely due to the text overlay.

*Management
keeps
prioritizing silly
rubbish that
stops people
from doing
their jobs*

*And what about project
managers ...*



Why is it important to ensure that you
don't do these things?

Maintainability



Readability

```
0571 :null, customfield_12630 :null, customfield_10575 :null, customfield_11930 :null, customfield_10723 :null, aggregateTimeestimate :null, creator":{"self":"https://jira.atlassian.com/rest/api/2/user?username=khannon", "name":"khannon", "key":"khannon", "emailAddress":"khannon at atlassian dot com", "avatarUrls":{"48x48":"https://jira.atlassian.com/secure/useravatar?avatarId=10612", "24x24":"https://jira.atlassian.com/secure/useravatar?size=small&avatarId=10612", "16x16":"https://jira.atlassian.com/secure/useravatar?size=xsmall&avatarId=10612", "32x32":"https://jira.atlassian.com/secure/useravatar?size=medium&avatarId=10612"}}, "displayName":"Keri Hannon [Atlassian]", "active":true, "timeZone":"America/Chicago"}, "customfield_15331":null, "subtasks":[], "customfield_10160":"950400", "customfield_10161":"true", "customfield_11130":"9223372036854775807", "customfield_13430":null, "customfield_10680":null, "reporter":{"self":"https://jira.atlassian.com/rest/api/2/user?username=khannon", "name":"khannon", "key":"khannon", "emailAddress":"khannon at atlassian dot com", "avatarUrls":{"48x48":"https://jira.atlassian.com/secure/useravatar?avatarId=10612", "24x24":"https://jira.atlassian.com/secure/useravatar?size=small&avatarId=10612", "16x16":"https://jira.atlassian.com/secure/useravatar?size=xsmall&avatarId=10612", "32x32":"https://jira.atlassian.com/secure/useravatar?size=medium&avatarId=10612"}}, "displayName":"Keri Hannon [Atlassian]", "active":true, "timeZone":"America/Chicago"}, "aggregateprogress":{"progress":0, "total":0}, "customfield_11531":null, "progress":{"progress":0, "total":0}, "votes":{"self":"https://jira.atlassian.com/rest/api/2/issue/JRA-45674/votes", "votes":0, "hasVoted":false}, "issuetype":{"self":"https://jira.atlassian.com/rest/api/2/issuetype/1", "id":"1", "description":"A problem which impairs or prevents the functions of the product.", "iconUrl":"https://jira.atlassian.com/secure/viewavatar?size=xsmall&avatarId=51493&avatarType=issuetype", "name":"Bug", "subtask":false, "avatarId":51493}, "timespent":null, "customfield_10150":["bmagro(bmagro)", "khannon(khannon)", "tony.squier@oracle.com(tony.squier@oracle.com)"], "project":{"self":"https://jira.atlassian.com/rest/api/2/project/10240", "id":"10240", "key":"JIRA", "name":"JIRA (including JIRA Core)", "avatarUrls":{"48x48":"https://jira.atlassian.com/secure/projectavatar?pid=10240&avatarId=17294", "24x24":"https://jira.atlassian.com/secure/projectavatar?size=small&pid=10240&avatarId=17294", "16x16":"https://jira.atlassian.com/secure/projectavatar?size=xsmall&pid=10240&avatarId=17294", "32x32":"https://jira.atlassian.com/secure/projectavatar?size=medium&pid=10240&avatarId=17294"}}, "projectCategory":{"self":"https://jira.atlassian.com/rest/api/2/projectCategory/10031", "id":"10031", "description":"","name":"Atlassian Products"}}, "customfield_12730":null, "aggregateTimespent":null, "resolutiondate":"2015-10-08T05:39:43.000+0000", "workratio":-1, "watches":{"self":"https://jira.atlassian.com/rest/api/2/issue/JRA-45674/watchers", "watchCount":2, "isWatching":false}, "customfield_15430":null, "created":"2015-10-07T19:00:34.000+0000", "customfield_11230":null, "customfield_15830":null, "customfield_11631":null, "customfield_12833":null, "customfield_12832":null, "customfield_14735":"0|11tyfk:", "updated":"2015-10-08T05:39:43.000+0000", "customfield_14330":null, "timeoriginalestimate":null, "customfield_10250":"<div class=\"\">

Atlassian is quite unique in that we allow all end users to create bugs / feature requests directly, without having to go through a support system.



Recently, many issues have been raised in this system, which are really support requests. This makes it harder for our support staff to provide the legendary support that we are famous for.



Bugs are reproducible, defined problems that we can get our development team to fix. If you are unsure of whether your problem is a bug or support request, please create a 

Thanks for your help. If you have any questions regarding this, please contact me directly.



Cheers,



Scott Farquhar



icon-info\"></span>


```



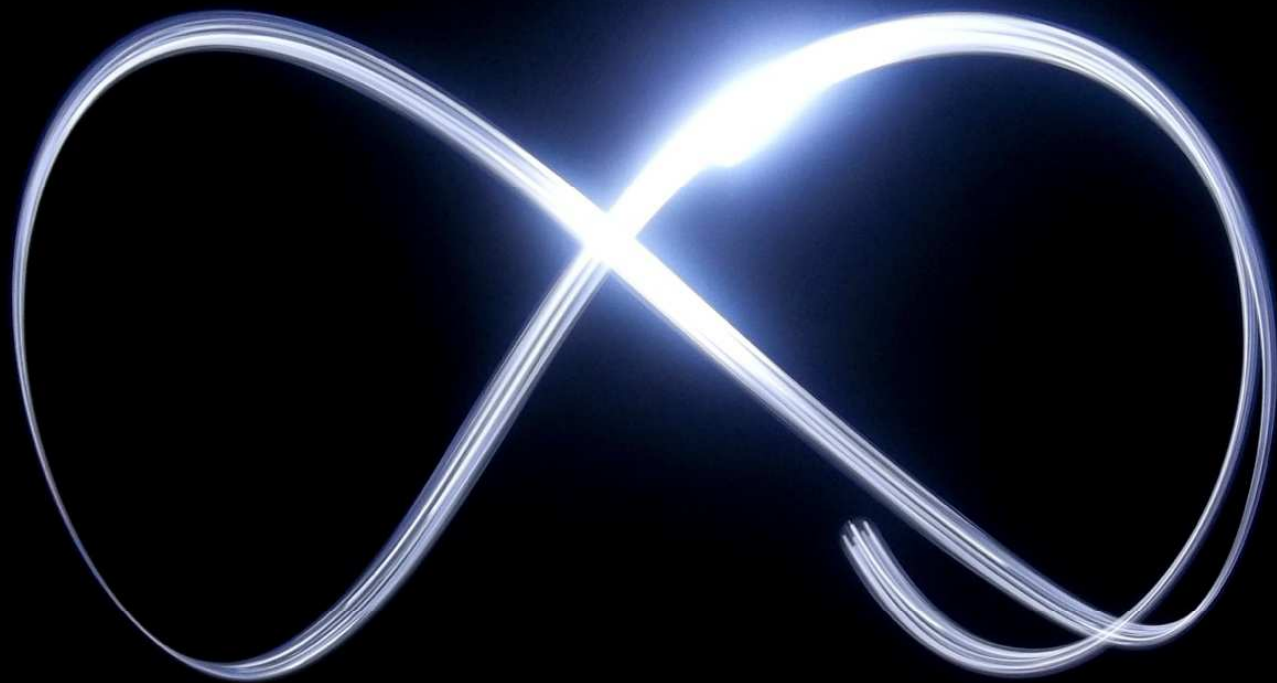

Extensibility

Upgradability



Supportability





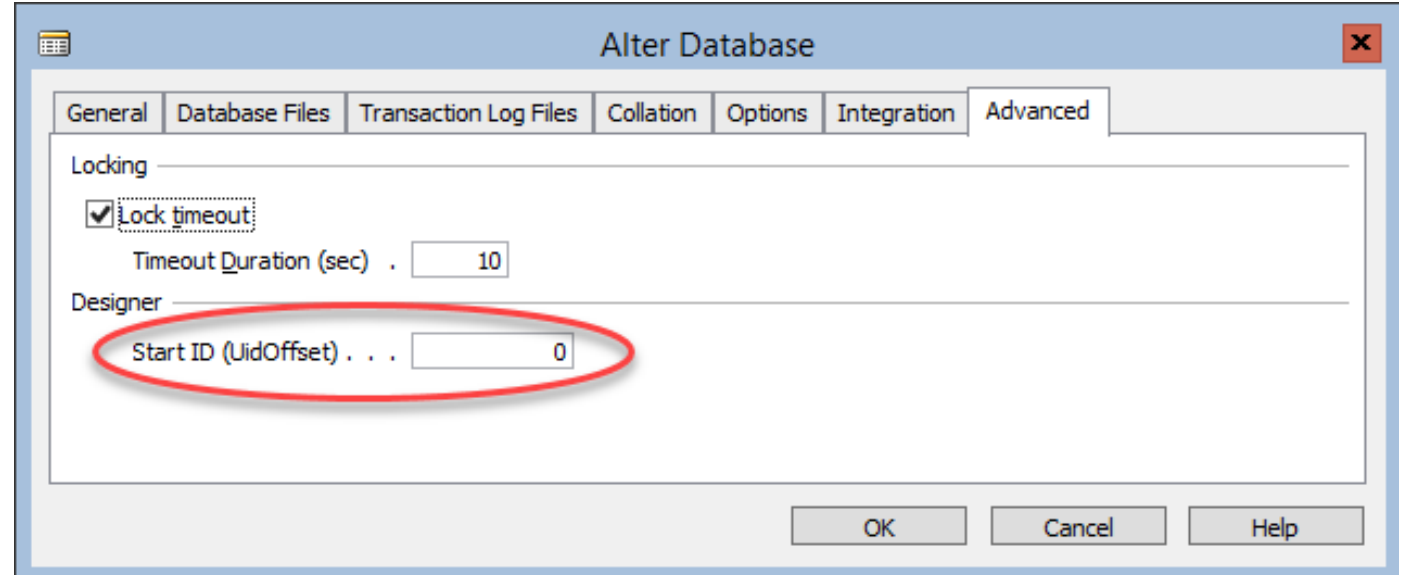
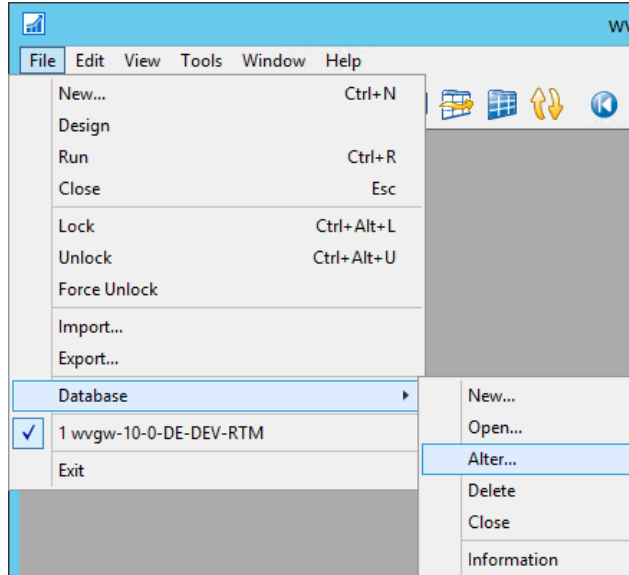
Bad Habits



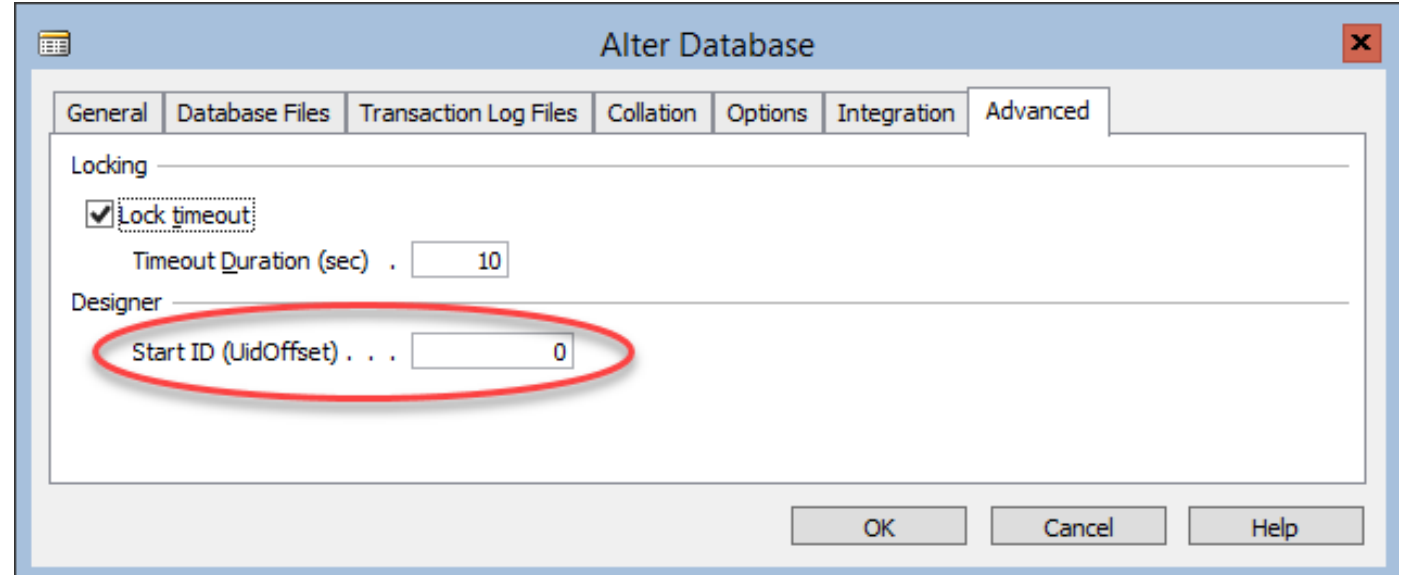
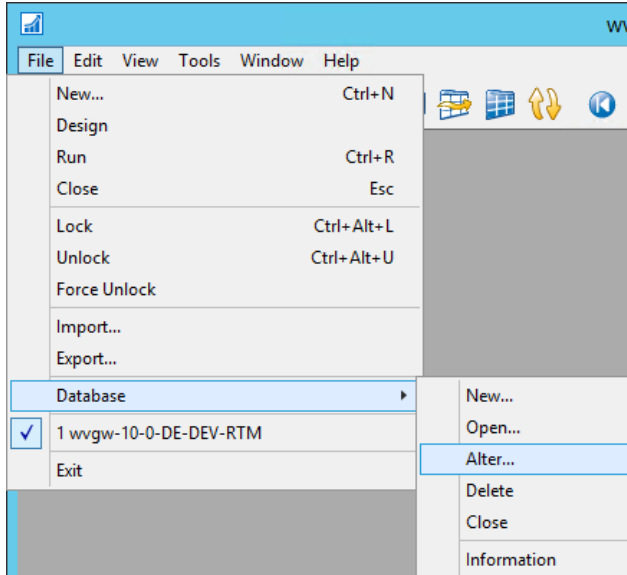
Start ID (UidOffset)



Who Needs the “Start ID (UidOffset)”?



Who Needs the “Start ID (UidOffset)”?



Everybody all the time.

Set it to the lowest integer of your object number range:

In your vertical, e.g.: 5,1 10,300

In your Dynamics 365 for Financials app, e.g.: 70,200,000

In your project objects, e.g.: 80,000

In your customers' project objects: 50,000

God Objects



What is a God Object

Object that knows or does too much

Holds too much data

Has too many methods

Typical God Objects in NAV

Codeunits

Tables

Functions

What is this?

[illegible]

Answer

OnRun trigger from Codeunit 80
All 1247 lines of it
In font size 2

God Table

A complex “generic” table that handles many different use cases
Typically starts as a “generic lookup” table...
... And then develops into a Frankenstein



God Table

Table 0 - Table Designer

E...Field No.	Field Name	Data Type	Length
1	Entity Table ID	Integer	
2	Entity No.	Code	20
3	Lookup Value	Text	30

Help

Table 0 - Table Designer

Field No.	Field Name	Data Type	Length	Des
1	Entity Table ID	Integer		
2	Entity No.	Code	20	
3	Lookup Value	Text	30	
4	Entity Type	Integer		
5	Entity Line No.	Integer		

Help

Table 0 - Table Designer

E...Field No.	Field Name	Data Type	Length
1	Entity Table ID	Integer	
2	Entity No.	Code	20
3	Lookup Value	Text	30
4	Entity Type	Integer	
5	Entity Line No.	Integer	
6	Attribute 1 Code	Code	20
7	Attribute 1 Value	Code	20
8	Attribute 2 Code	Code	20
9	Attribute 2 Value	Code	20
10	Attribute 3 Code	Code	20
11	Attribute 3 Value	Code	20
12	Entity Status	Integer	
13	Template Name	Code	10
14	Batch Name	Code	10
15	Attribute 2 Code	Code	20
16	Attribute 2 Value	Code	20
17	Attribute 3 Code	Code	20
18	Attribute 3 Value	Code	20

Help

Good principles

Separation of concerns

One table represents one entity

No “generic” attributes

80-line rule



Collection Codeunits



Collection Codeunits

Do not turn codeunits into functional collections like:
Everything to do with Sales Conditions
Everything to do with Freight Management
etc.



A codeunit must represent one method – and one method only.

Code Cloning



Code Cloning

CU 229
Document-Print
NAV 2016

```
PrintPurchHeader(PurchHeader : Record "Purchase Header")
PurchHeader.SETRANGE("No.",PurchHeader."No.");
PurchSetup.GET;
IF PurchSetup."Calc. Inv. Discount" THEN BEGIN
    PurchLine.RESET;
    PurchLine.SETRANGE("Document Type",PurchHeader."Document Type");
    PurchLine.SETRANGE("Document No.",PurchHeader."No.");
    PurchLine.FINDFIRST;
    PurchCalcDisc.RUN(PurchLine);
    PurchHeader.GET(PurchHeader."Document Type",PurchHeader."No.");
    COMMIT;
END;
CASE PurchHeader."Document Type" OF
    PurchHeader."Document Type"::Quote:
        ReportSelection.SETRANGE(Usage,ReportSelection.Usage::"P.Quote");
    PurchHeader."Document Type"::"Blanket Order":
        ReportSelection.SETRANGE(Usage,ReportSelection.Usage::"P.Blanket");
    PurchHeader."Document Type"::Order:
        ReportSelection.SETRANGE(Usage,ReportSelection.Usage::"P.Order");
    PurchHeader."Document Type"::"Return Order":
        ReportSelection.SETRANGE(Usage,ReportSelection.Usage::"P.Return");
ELSE
    EXIT;
END;
ReportSelection.SETFILTER("Report ID",'<>0');
ReportSelection.FIND('-');
REPEAT
    REPORT.RUNMODAL(ReportSelection."Report ID",TRUE,FALSE,PurchHeader)
UNTIL ReportSelection.NEXT = 0;
```

Code Cloning

CU 229
Document-Print
NAV 2016

```
PrintServiceHeader(ServiceHeader : Record "Service Header")
SalesSetup.GET;
ServiceHeader.SETRANGE("No.",ServiceHeader."No.");
IF SalesSetup."Calc. Inv. Discount" THEN BEGIN
    ServLine.RESET;
    ServLine.SETRANGE("Document Type",ServiceHeader."Document Type");
    ServLine.SETRANGE("Document No.",ServiceHeader."No.");
    ServLine.FINDFIRST;
    ServCalcDisc.RUN(ServLine);
    ServiceHeader.GET(ServiceHeader."Document Type",ServiceHeader."No.");
    COMMIT;
END;
CASE ServiceHeader."Document Type" OF
    ServiceHeader."Document Type"::Quote:
        ReportSelection.SETRANGE(Usage,ReportSelection.Usage::"SM.Quote");
    ServiceHeader."Document Type"::Order:
        ReportSelection.SETRANGE(Usage,ReportSelection.Usage::"SM.Order");
    ServiceHeader."Document Type"::Invoice:
        ReportSelection.SETRANGE(Usage,ReportSelection.Usage::"SM.Invoice");
    ServiceHeader."Document Type"::"Credit Memo":
        ReportSelection.SETRANGE(Usage,ReportSelection.Usage::"SM.Credit Memo");
ELSE
    EXIT;
END;
ReportSelection.SETFILTER("Report ID",<>0');
IF ReportSelection.FIND('-') THEN BEGIN
    REPEAT
        REPORT.RUNMODAL(ReportSelection."Report ID",TRUE,FALSE,ServiceHeader);
    UNTIL ReportSelection.NEXT = 0;
END ELSE
    ERROR(Text002,ReportSelection.FIELDCAPTION("Report ID"),
        ServiceHeader.TABLECAPTION,ReportSelection.TABLECAPTION);
```

Code Cloning

[illegible][illegible][illegible][illegible]

Code Cloning



mibuso.com

```

    @RequestMapping("search") @Valid @ResponseBody
    public Response search(@RequestParam("q") String q) {
        return Response.ok(new ArrayList<Book>()).build();
    }

    @RequestMapping("add")
    public Response add(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
    }

    @RequestMapping("update")
    public Response update(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
    }

    @RequestMapping("delete")
    public Response delete(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
    }

    @RequestMapping("get")
    public Response get(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
    }

    @RequestMapping("list")
    public Response list(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
    }

    @RequestMapping("listAll")
    public Response listAll() {
        return Response.ok(new ArrayList<Book>()).build();
    }

    @RequestMapping("listByGenre")
    public Response listByGenre(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
    }

    @RequestMapping("listByAuthor")
    public Response listByAuthor(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
    }

    @RequestMapping("listByYear")
    public Response listByYear(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
    }

    @RequestMapping("listByTitle")
    public Response listByTitle(@Valid @RequestBody Book book) {
        return Response.ok(book).build();
   }
}

```

[illegible]

```
ReportTransHeader(transheader : ReportTransHeader)
{
  transheader.SETNAME("No.", transheader."No.");
  ReportSelectList.SETNAME(transheader."Name");
  ReportSelectList.SET("NAME", transheader."NAME");
  ReportSelectList.FIND("-");
}
REPORT
REPORT NAME(transheader."Report", transheader."NAME", transheader."NAME");
REPORT REPORT NAME + 1;
```

[illegible][illegible][illegible][illegible][illegible][illegible][illegible]

```

15 ReportSelections.Delete; Print(ComboBox1.SelectedItem, SelectedMail, FN); <= NEW MSG
16 ReportSelections.Add(FN, SelectedMail, "Report ID", "v");
17 ReportSelections.Remove(FN);
18 MSG;
19
20 [SelectedMail] THEN MSG;
21 Attachments.AddTo SelectedMail.ReportID(SelectedMail.ReportSelections, "Report ID");
22 Deattachments.Add(SelectedMail.SelectedMail, Attachments.AddTo, SelectedMail);
23 MSG;
24
25 REPORT: RptReportSelections, "Report ID", MSG, AGM, SelectedMail;
26 [MSG] ReportSelections.Next <= FN;
27 MSG;

```

[illegible][illegible][illegible][illegible][illegible][illegible]

```

103     if (hasField(1)) THEN {
104         attachNext(1).Label := AttachNextLabel(AttachNextLabel, AttachNextLabel, "New
105         AttachNextLabel(AttachNextLabel, AttachNextLabel, AttachNextLabel, AttachNextLabel)
106     }
107     RETURN AttachNextLabel(AttachNextLabel, "Report", "ID", "URL", "URL", AttachNextLabel)
108 }
109
110
111 AttachNextLabel :=
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
```

[illegible][illegible][illegible][illegible][illegible]

Code Cloning

Be as generic as possible

RecordRef/FieldRef

Variant Façade Pattern

<http://community.dynamics.com/nav/w/designpatterns/246.variant-facade>

Polymorphic Design Patterns by Vjeko

<http://vjeko.com>

Code Cloning

```
PrintPurchHeader(PurchHeader : Record "Purchase Header")  
PurchHeader.SETRANGE("No.",PurchHeader."No.");  
CalcPurchDisc(PurchHeader);  
  
ReportSelections.Print(GetPurchDocTypeUsage(PurchHeader),PurchHeader,'');
```


Code Cloning

```
PrintPurchHeader(PurchHeader : Record "Purchase Header")  
PurchHeader.SETRANGE("No.",PurchHeader."No.");  
CalcPurchDisc(PurchHeader);  
  
ReportSelections.Print(GetPurchDocTypeUsage(PurchHeader),PurchHeader,"");
```

```
Print(ReportUsage : Integer;RecordVariant : Variant;CustNo : Code[20])  
PrintWithGUIYesNo(ReportUsage,RecordVariant,TRUE,CustNo);
```

Code Cloning

```
PrintPurchHeader(PurchHeader : Record "Purchase Header")  
PurchHeader.SETRANGE("No.",PurchHeader."No.");  
CalcPurchDisc(PurchHeader);  
  
ReportSelections.Print(GetPurchDocTypeUsage(PurchHeader),PurchHeader,"");
```

```
Print(ReportUsage : Integer;RecordVariant : Variant;CustNo : Code[20])  
PrintWithGUIYesNo(ReportUsage,RecordVariant,TRUE,CustNo);
```

```
PrintWithGUIYesNo(ReportUsage : Integer;RecordVariant : Variant;IsGUI : Boolean;CustNo : Code[20])  
FindPrintUsage(ReportUsage,CustNo,TempReportSelections);  
WITH TempReportSelections DO  
    REPEAT  
        ReportLayoutSelection.SetTempLayoutSelected("Custom Report Layout Code");  
        REPORT.RUNMODAL("Report ID",IsGUI,FALSE,RecordVariant)  
    UNTIL NEXT = 0;  
ReportLayoutSelection.SetTempLayoutSelected('');
```



Code on Pages

Code on Pages



Page 42 Sales Order - C/AL Editor

```
89 CalculateFreightCost - OnAction()
90 // MyVertical.s
91 IF NOT CONFIRM(CnfrmCalcFreightCost,FALSE,"Document Type","No.") THEN
92     EXIT;
93
94 SalesLine.SETCURRENTKEY("Document Type","Document No.",Type,"No.");
95 SalesLine.SETRANGE("Document Type","Document Type");
96 // Lots of more business logic code
97 // MyVertical.e
```

Code on Pages



Page 42 Sales Order - C/AL Editor

```
88 |
89 | CtrlCalculateFreightCost - OnAction()
90 | // MyVertical.s
91 | IF NOT CONFIRM(CnfrmCalcFreightCost,FALSE,"Document Type","No.") THEN
92 |     EXIT;
93 |
94 | SalesLine.SETCURRENTKEY("Document Type","Document No.",Type,"No.");
95 | SalesLine.SETRANGE("Document Type","Document Type");
96 |
97 | // Lots of more business logic code
98 |
99 | // MyVertical.e
```


Code on Pages

```
Page 42 Sales Order - C/AL Editor
89 CtrlCalculateFreightCost - OnAction()
90 // MyVertical.s
91 ThisCalculateFreightCost;
92 // MyVertical.e
```

```
Page 42 Sales Order - C/AL Editor
970
971 LOCAL ThisCalculateFreightCost()
972 // MyVertical.s
973 CalculateFreightCost;
974 // MyVertical.e
```

OnAction
Trigger

Calls Local
Function on
Page

Calls Method
Declaration
on Class

Calls Method

Code on Pages



Page 42 Sales Order - C/AL Editor

```
30 |  
31 | OnAfterGetRecord()  
32 |   SetControlVisibility;  
33 |   UpdateShipToBillToGroupVisibility;  
34 |
```



Page 46 Sales Order Subform - C/AL Editor

```
803 |  
804 | <Control1300> - OnValidate()  
805 |   ValidateShortcutDimCode(3,ShortcutDimCode[3]);
```

Code on Pages

Business logic must not be coded on a page.

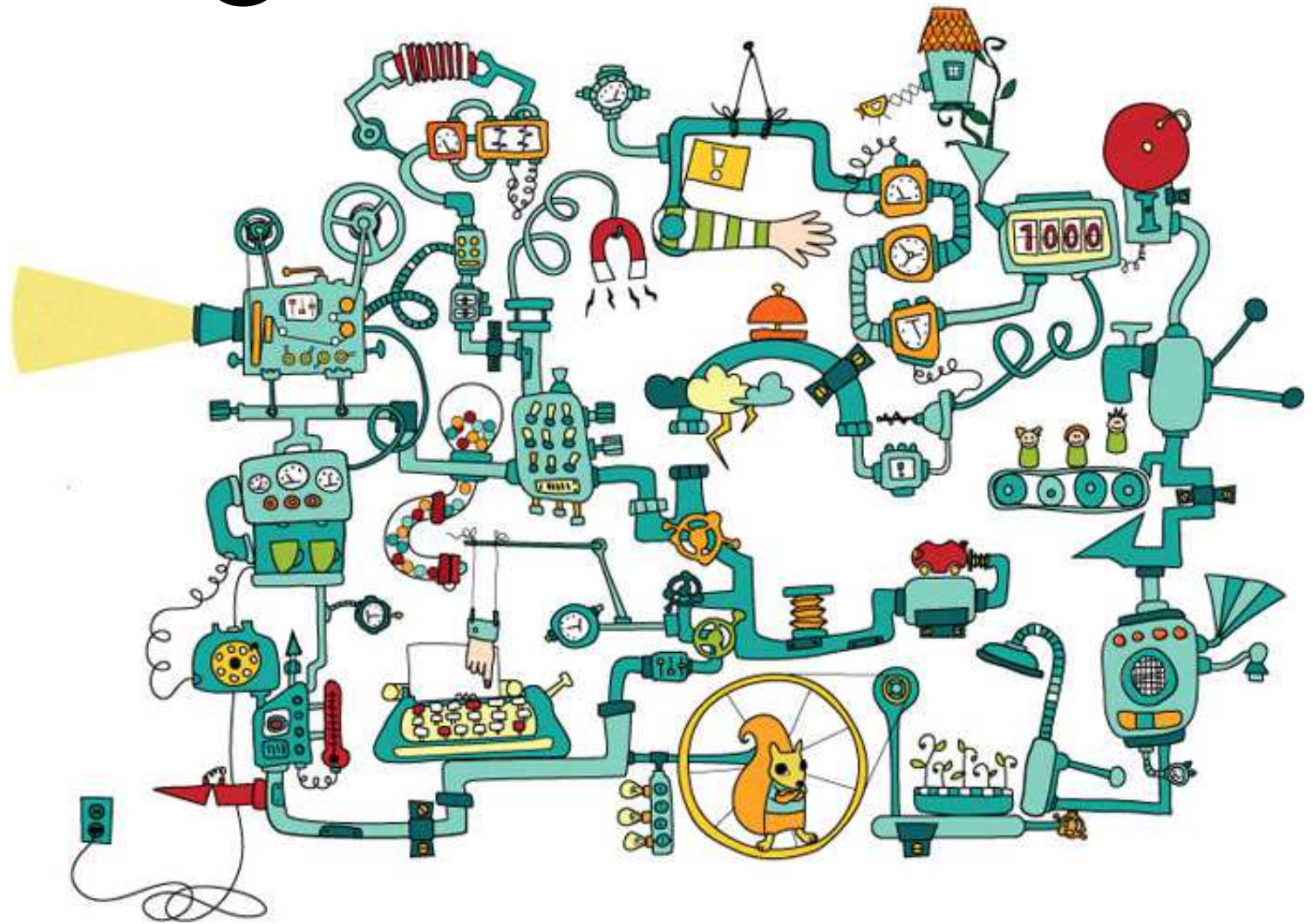
Avoid any code in OnAction triggers, if it does more than pointing to a local function.

Code which triggers business logic by invoking a method on a class is fine.

In a few contexts, especially when passing temporary records, code which sets or returns parameters is fine.

Code which manipulates the UI is fine.

Refactoring



Decipher this, and I buy you lunch

```
IF Total < 0 THEN
  IF ModuleSetup.GET THEN
    IF (Entity.Rounding <> '') AND (ModuleSetup."Amount Rounding Precision" > 0) THEN BEGIN
      Decimal := ABS(Total) - ROUND(ABS(Total),1,'<');
      Counter := 0;
      Return := Counter * ModuleSetup."Amount Rounding Precision" - Decimal;
      REPEAT
        Counter := Counter + 1;
        IF ABS(Counter * ModuleSetup."Amount Rounding Precision" - Decimal) <= ABS(Return) THEN
          Return := Counter * ModuleSetup."Amount Rounding Precision" - Decimal;
      UNTIL Counter * ModuleSetup."Amount Rounding Precision" >= 1;
    END;
```