

BADI's, User-exits, Enhancements... como encontrá-los?

19/07/2012 14:00

Faaaaa Zumbizada! Estou de volta no modo "zombie = ON" por causa do projeto, testes integrados que estão acabando com a minha vida (e que nunca acabam!) 😞

Bom, mas chega de mimimi e vamos lá!!! Neste post vou falar sobre alguns métodos para encontrar Enhancements, User-exits, BADI's e etc. Aposto que isso pode ajudar muito, principalmente para os zumbis mais novinhos. 😊



Não, isso não é uma user-exit...

CALL CUSTOMER FUNCTION

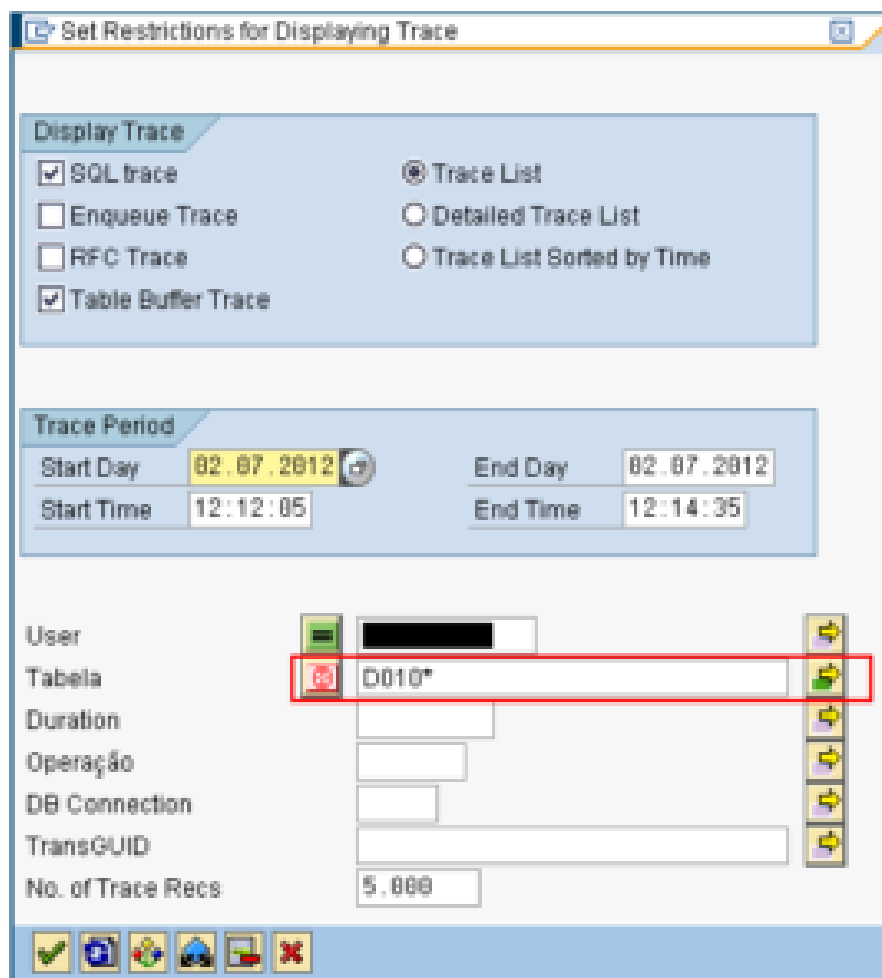
Você pode colocar um BREAK-POINT nesse comando, que é usado para chamar as User-exits . Já falamos dele aqui no blog e se quiser saber mais, [clique aqui](#).

BADI'S

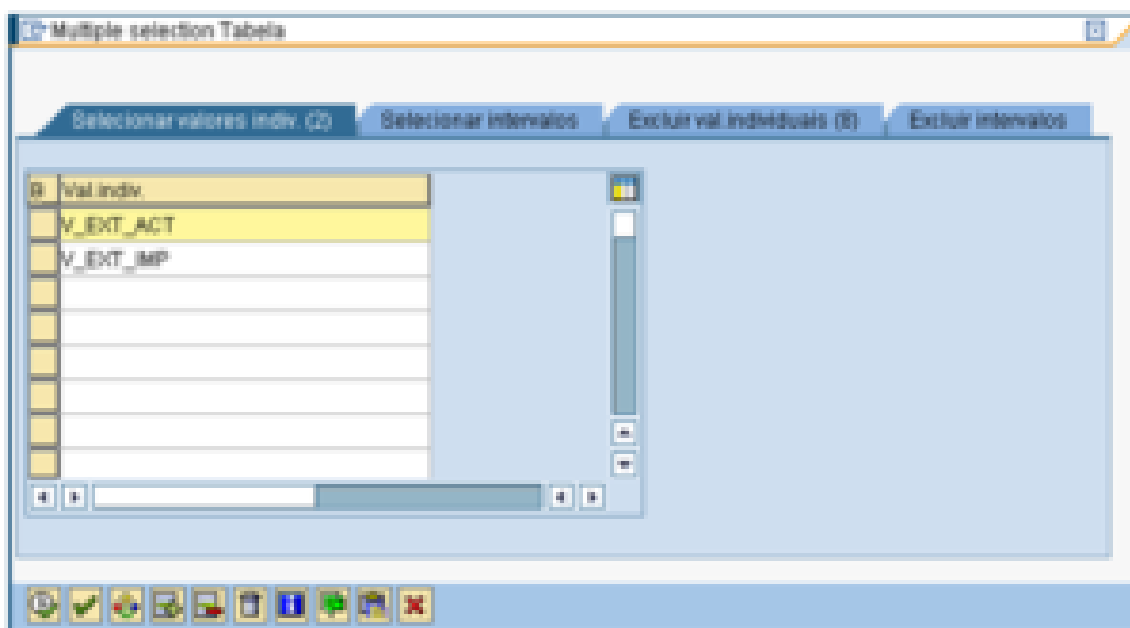
Método 1: V_EXT_ACT e V_EXT_IMP

A BADI faz uma seleção nas tabelas de visões V_EXT_ACT e V_EXT_IMP. Então podemos ir na transação ST05 -> Selecionar "SQL Trace" e "Table Buffer Trace" -> Activate Trace -> Executar a transação -> Deactivate Trace -> Display Trace.

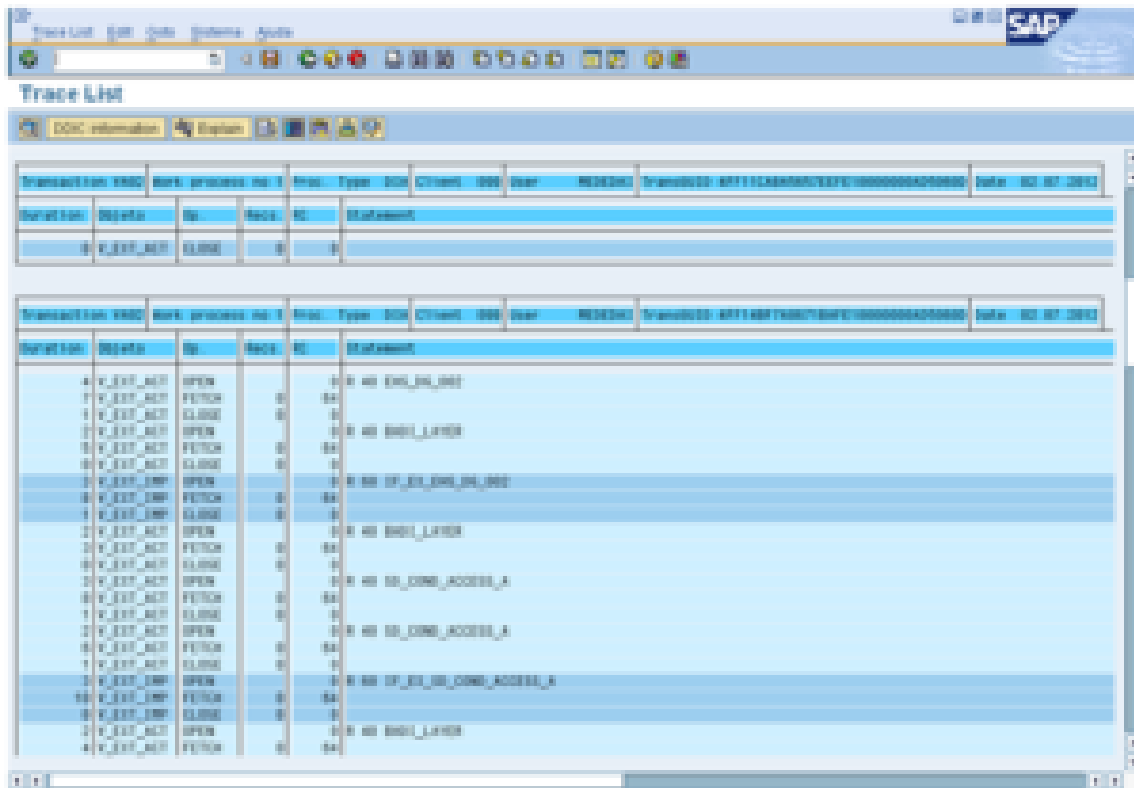
Irá aparecer a tela abaixo:



Vamos seleccionar os seguintes objetos:



No resultado final irá mostrar todas as BADI's na ordem em que elas foram executadas:



The screenshot shows the SAP Trace List interface. The top bar includes 'Trace List', 'Edit', 'Tools', 'System', and 'Help'. Below the toolbar, the 'Trace List' title is displayed. The main area shows a table of transaction data and a detailed list of statements.

Transaction	Work process no.	Proc. Type	Stat.	Client	User	Message	Transaction	Date
SE24	1	0	0	0	0	0	SE24	02.07.2012

Serial No.	Object	No.	Recs.	Stat.	Statement
1	CL_EXITHANDLER	0	0	0	

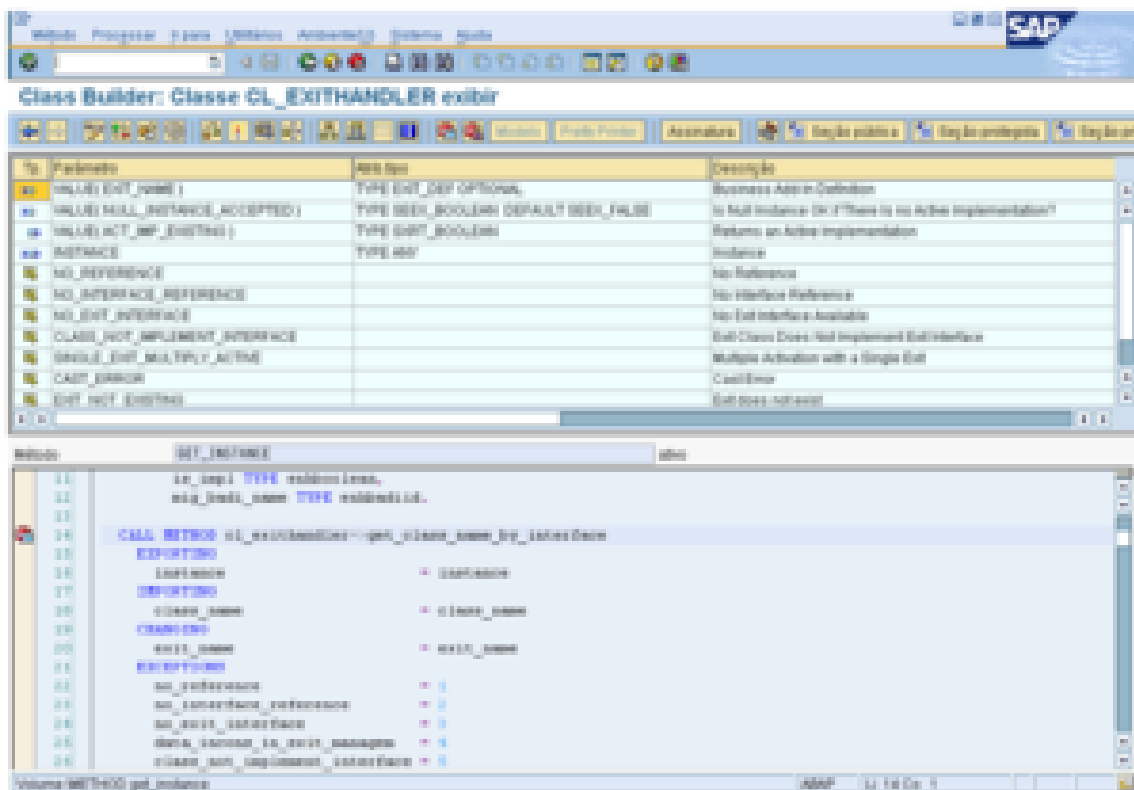
Transaction	Work process no.	Proc. Type	Stat.	Client	User	Message	Transaction	Date
SE24	1	0	0	0	0	0	SE24	02.07.2012

Serial No.	Object	No.	Recs.	Stat.	Statement
1	CL_EXITHANDLER	0	0	0	
2	CL_EXITHANDLER	0	0	0	
3	CL_EXITHANDLER	0	0	0	
4	CL_EXITHANDLER	0	0	0	
5	CL_EXITHANDLER	0	0	0	
6	CL_EXITHANDLER	0	0	0	
7	CL_EXITHANDLER	0	0	0	
8	CL_EXITHANDLER	0	0	0	
9	CL_EXITHANDLER	0	0	0	
10	CL_EXITHANDLER	0	0	0	
11	CL_EXITHANDLER	0	0	0	
12	CL_EXITHANDLER	0	0	0	
13	CL_EXITHANDLER	0	0	0	
14	CL_EXITHANDLER	0	0	0	
15	CL_EXITHANDLER	0	0	0	
16	CL_EXITHANDLER	0	0	0	
17	CL_EXITHANDLER	0	0	0	
18	CL_EXITHANDLER	0	0	0	
19	CL_EXITHANDLER	0	0	0	
20	CL_EXITHANDLER	0	0	0	
21	CL_EXITHANDLER	0	0	0	
22	CL_EXITHANDLER	0	0	0	
23	CL_EXITHANDLER	0	0	0	
24	CL_EXITHANDLER	0	0	0	
25	CL_EXITHANDLER	0	0	0	
26	CL_EXITHANDLER	0	0	0	
27	CL_EXITHANDLER	0	0	0	
28	CL_EXITHANDLER	0	0	0	
29	CL_EXITHANDLER	0	0	0	
30	CL_EXITHANDLER	0	0	0	
31	CL_EXITHANDLER	0	0	0	
32	CL_EXITHANDLER	0	0	0	
33	CL_EXITHANDLER	0	0	0	
34	CL_EXITHANDLER	0	0	0	
35	CL_EXITHANDLER	0	0	0	
36	CL_EXITHANDLER	0	0	0	
37	CL_EXITHANDLER	0	0	0	
38	CL_EXITHANDLER	0	0	0	
39	CL_EXITHANDLER	0	0	0	
40	CL_EXITHANDLER	0	0	0	
41	CL_EXITHANDLER	0	0	0	
42	CL_EXITHANDLER	0	0	0	
43	CL_EXITHANDLER	0	0	0	
44	CL_EXITHANDLER	0	0	0	
45	CL_EXITHANDLER	0	0	0	
46	CL_EXITHANDLER	0	0	0	
47	CL_EXITHANDLER	0	0	0	
48	CL_EXITHANDLER	0	0	0	
49	CL_EXITHANDLER	0	0	0	
50	CL_EXITHANDLER	0	0	0	
51	CL_EXITHANDLER	0	0	0	
52	CL_EXITHANDLER	0	0	0	
53	CL_EXITHANDLER	0	0	0	
54	CL_EXITHANDLER	0	0	0	
55	CL_EXITHANDLER	0	0	0	
56	CL_EXITHANDLER	0	0	0	
57	CL_EXITHANDLER	0	0	0	
58	CL_EXITHANDLER	0	0	0	
59	CL_EXITHANDLER	0	0	0	
60	CL_EXITHANDLER	0	0	0	
61	CL_EXITHANDLER	0	0	0	
62	CL_EXITHANDLER	0	0	0	
63	CL_EXITHANDLER	0	0	0	
64	CL_EXITHANDLER	0	0	0	
65	CL_EXITHANDLER	0	0	0	
66	CL_EXITHANDLER	0	0	0	
67	CL_EXITHANDLER	0	0	0	
68	CL_EXITHANDLER	0	0	0	
69	CL_EXITHANDLER	0	0	0	
70	CL_EXITHANDLER	0	0	0	
71	CL_EXITHANDLER	0	0	0	
72	CL_EXITHANDLER	0	0	0	
73	CL_EXITHANDLER	0	0	0	
74	CL_EXITHANDLER	0	0	0	
75	CL_EXITHANDLER	0	0	0	
76	CL_EXITHANDLER	0	0	0	
77	CL_EXITHANDLER	0	0	0	
78	CL_EXITHANDLER	0	0	0	
79	CL_EXITHANDLER	0	0	0	
80	CL_EXITHANDLER	0	0	0	
81	CL_EXITHANDLER	0	0	0	
82	CL_EXITHANDLER	0	0	0	
83	CL_EXITHANDLER	0	0	0	
84	CL_EXITHANDLER	0	0	0	
85	CL_EXITHANDLER	0	0	0	
86	CL_EXITHANDLER	0	0	0	
87	CL_EXITHANDLER	0	0	0	
88	CL_EXITHANDLER	0	0	0	
89	CL_EXITHANDLER	0	0	0	
90	CL_EXITHANDLER	0	0	0	
91	CL_EXITHANDLER	0	0	0	
92	CL_EXITHANDLER	0	0	0	
93	CL_EXITHANDLER	0	0	0	
94	CL_EXITHANDLER	0	0	0	
95	CL_EXITHANDLER	0	0	0	
96	CL_EXITHANDLER	0	0	0	
97	CL_EXITHANDLER	0	0	0	
98	CL_EXITHANDLER	0	0	0	
99	CL_EXITHANDLER	0	0	0	
100	CL_EXITHANDLER	0	0	0	

Método 2: CL_EXITHANDLER=>GET_CLASS_NAME_BY_INTERFACE

Na transação SE24, vamos selecionar a classe CL_EXITHANDLER, ir para o método GET_INSTANCE e colocar um BREAK-POINT.

Através do parâmetro de exportação EXIT_NAME é possível saber o nome da BADI que será chamada.



The screenshot shows the SAP Class Builder interface for the class CL_EXITHANDLER. The top bar includes 'Method', 'Program', 'Class', 'Attributes', 'System', and 'Help'. Below the toolbar, the 'Class Builder: Classe CL_EXITHANDLER exibir' title is displayed. The main area shows a table of class attributes and a detailed view of the GET_INSTANCE method.

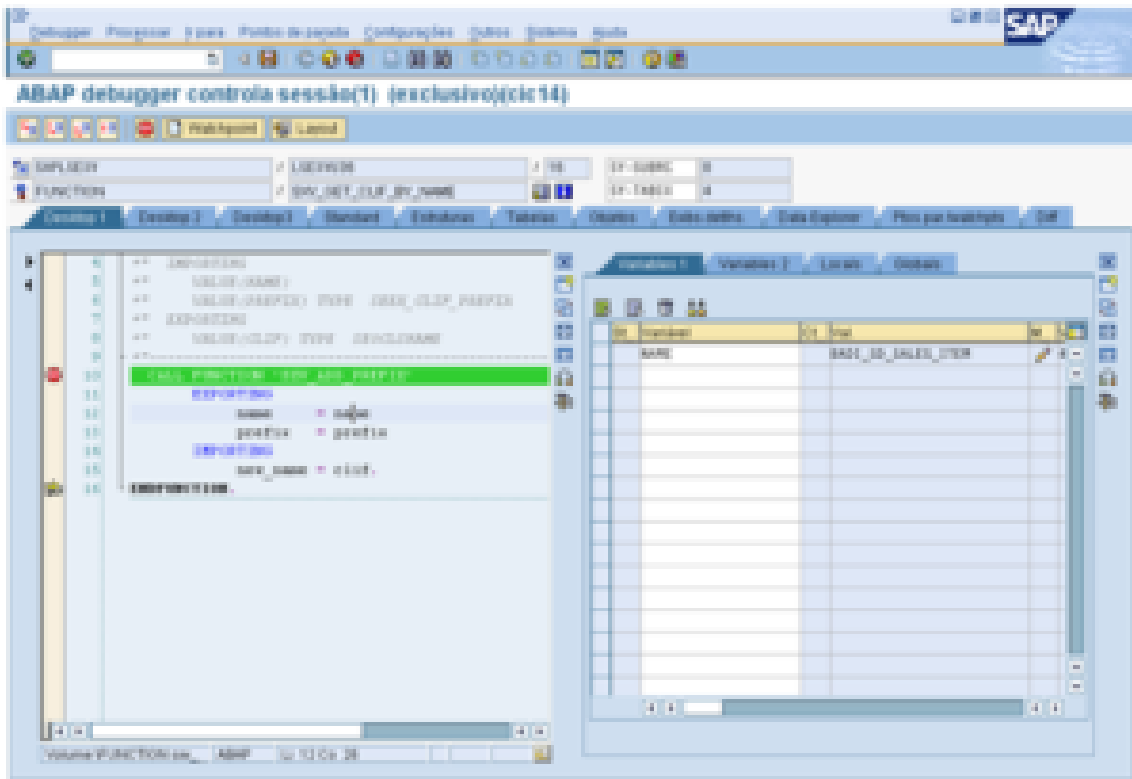
To	Parâmetros	Atributos	Descrição
1	VALUE EXIT_NAME	TYPE EXIT_NAME OPTIONAL	Business Application
2	VALUE MULT_INSTANCE_ACCEPTED	TYPE SEEN_BOOLEAN DEFAULT SEEN_FALSE	Is Mult Instance? Or if there is no Active Implementation?
3	VALUE ACT_INF_EXITING	TYPE SEEN_BOOLEAN	Returns an Active Implementation
4	INSTANCE	TYPE REF	Instance
5	NO_REFERENCE		No Reference
6	NO_INTERFACE_REFERENCE		No Interface Reference
7	NO_EXIT_INTERFACE		No Exit Interface Available
8	CLASS_NOT_IMPLEMENT_INTERFACE		Exit Class Does Not Implement Exit Interface
9	SEEN_EXIT_MULTIPLE_ACTIVE		Multiple Activation with a Single Exit
10	CASE_ERROR		Case Error
11	EXIT_NOT_EXITING		Exit does not exist

Method	GET_INSTANCE	Attributes
1	EXIT_NAME TYPE EXIT_NAME	
2	MULT_INSTANCE_ACCEPTED TYPE SEEN_BOOLEAN	
3	CALL METHOD cl_exithandler~get_class_name_by_interface	
4	EXPORTING	
5	EXIT_NAME	
6	CLASS_NAME	
7	CLASS_NAME	
8	EXIT_NAME	
9	EXIT_NAME	
10	EXIT_NAME	
11	EXIT_NAME	
12	EXIT_NAME	
13	EXIT_NAME	
14	EXIT_NAME	
15	EXIT_NAME	
16	EXIT_NAME	
17	EXIT_NAME	
18	EXIT_NAME	
19	EXIT_NAME	
20	EXIT_NAME	
21	EXIT_NAME	
22	EXIT_NAME	
23	EXIT_NAME	
24	EXIT_NAME	
25	EXIT_NAME	
26	EXIT_NAME	
27	EXIT_NAME	
28	EXIT_NAME	
29	EXIT_NAME	
30	EXIT_NAME	
31	EXIT_NAME	
32	EXIT_NAME	
33	EXIT_NAME	
34	EXIT_NAME	
35	EXIT_NAME	
36	EXIT_NAME	
37	EXIT_NAME	
38	EXIT_NAME	
39	EXIT_NAME	
40	EXIT_NAME	
41	EXIT_NAME	
42	EXIT_NAME	
43	EXIT_NAME	
44	EXIT_NAME	
45	EXIT_NAME	
46	EXIT_NAME	
47	EXIT_NAME	
48	EXIT_NAME	
49	EXIT_NAME	
50	EXIT_NAME	
51	EXIT_NAME	
52	EXIT_NAME	
53	EXIT_NAME	
54	EXIT_NAME	
55	EXIT_NAME	
56	EXIT_NAME	
57	EXIT_NAME	
58	EXIT_NAME	
59	EXIT_NAME	
60	EXIT_NAME	
61	EXIT_NAME	
62	EXIT_NAME	
63	EXIT_NAME	
64	EXIT_NAME	
65	EXIT_NAME	
66	EXIT_NAME	
67	EXIT_NAME	
68	EXIT_NAME	
69	EXIT_NAME	
70	EXIT_NAME	
71	EXIT_NAME	
72	EXIT_NAME	
73	EXIT_NAME	
74	EXIT_NAME	
75	EXIT_NAME	
76	EXIT_NAME	
77	EXIT_NAME	
78	EXIT_NAME	
79	EXIT_NAME	
80	EXIT_NAME	
81	EXIT_NAME	
82	EXIT_NAME	
83	EXIT_NAME	
84	EXIT_NAME	
85	EXIT_NAME	
86	EXIT_NAME	
87	EXIT_NAME	
88	EXIT_NAME	
89	EXIT_NAME	
90	EXIT_NAME	
91	EXIT_NAME	
92	EXIT_NAME	
93	EXIT_NAME	
94	EXIT_NAME	
95	EXIT_NAME	
96	EXIT_NAME	
97	EXIT_NAME	
98	EXIT_NAME	
99	EXIT_NAME	
100	EXIT_NAME	

Método 3: FM SXV_GET_CLIF_BY_NAME.

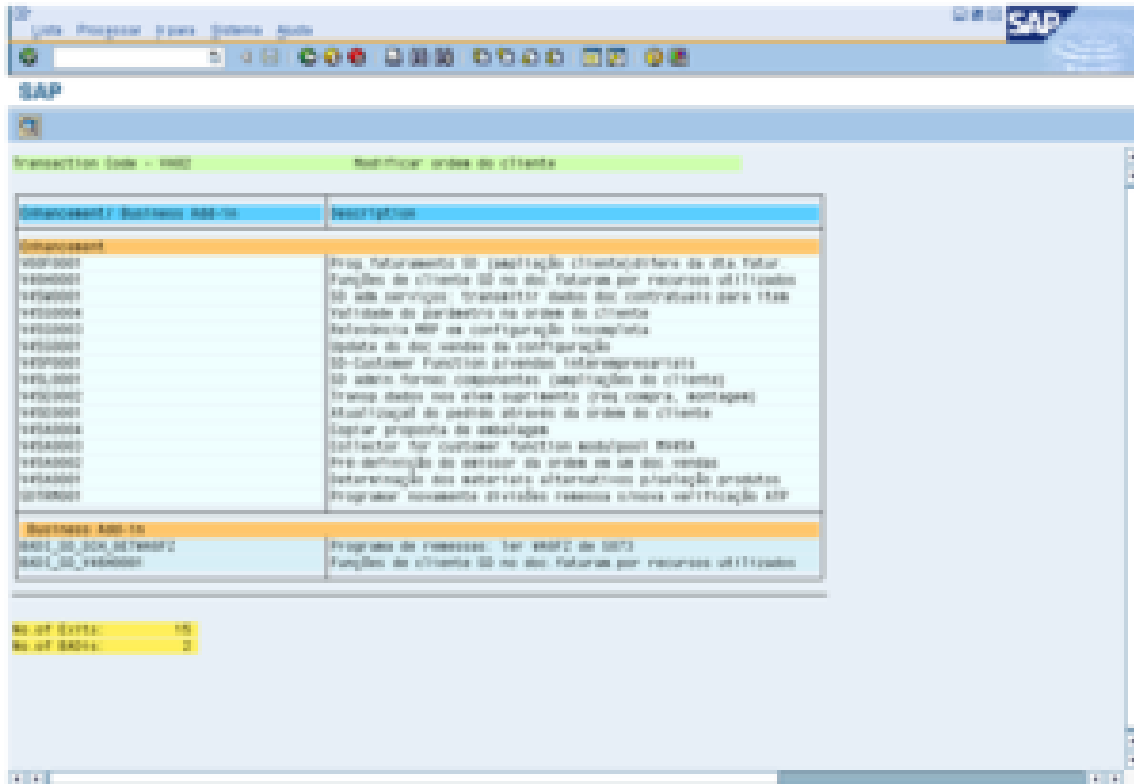
Um outro método é colocando um BREAK-POINT na função SXV_GET_CLIF_BY_NAME, que é chamada toda vez que alguma BADI é executada.

O parâmetro de exportação NAME irá mostrar o nome da BADI.



E aí, achou difícil, chato, deprimente? Não se assuste, vou mostrar algumas ferramentas que podem facilitar a sua vida! 😊

Há algum tempo atrás precisei encontrar uma user-exit de uma transação X, e no meio de um fórum no SCN encontrei um programa bastante útil! Ele lista todas as user-exits e BADI's de uma transação ou programa.



Abaixo está o código-fonte. Não sei quem é o autor e não será coincidência encontrar esse mesmo programa em outros fóruns (é bem manjado).

```
REPORT z_enhancement_finder NO STANDARD PAGE HEADING LINE-SIZE 201.
```

```
TABLES : tstc,
          tadir,
          modsapt,
          modact,
          trdir,
          tfdir,
          enlfdir,
          sxs_attrt ,
          tstct.
```

```
DATA : jtab LIKE tadir OCCURS 0 WITH HEADER LINE.
DATA : field1(30).
DATA : v_devclass LIKE tadir-devclass.
```

```
PARAMETERS : p_tcode LIKE tstc-tcode,
              p_pgmna LIKE tstc-pgmna .
```

```
DATA wa_tadir TYPE tadir.
```

```
START-OF-SELECTION.
```

```
IF NOT p_tcode IS INITIAL.
  SELECT SINGLE * FROM tstc WHERE tcode EQ p_tcode.
```

```
ELSEIF NOT p_pgmna IS INITIAL.
  tstc-pgmna = p_pgmna.
ENDIF.
```

```
IF sy-subrc EQ 0.
  SELECT SINGLE * FROM tadir
  WHERE pgmid = 'R3TR'
```

```

AND object = 'PROG'
AND obj_name = tstc-pgmna.

MOVE : tadir-devclass TO v_devclass.

IF sy-subrc NE 0.
  SELECT SINGLE * FROM trdir
  WHERE name = tstc-pgmna.
  IF trdir-subc EQ 'F'.
    SELECT SINGLE * FROM tfdir
    WHERE pname = tstc-pgmna.

    SELECT SINGLE * FROM enlfdir
    WHERE funcname = tfdir-funcname.

    SELECT SINGLE * FROM tadir
    WHERE pgmid = 'R3TR'
    AND object = 'FUGR'
    AND obj_name EQ enlfdir-area.

    MOVE : tadir-devclass TO v_devclass.
  ENDIF.
ENDIF.

SELECT * FROM tadir INTO TABLE jtab
WHERE pgmid = 'R3TR'
AND object IN ('SMOD', 'SXSD')
AND devclass = v_devclass.

SELECT SINGLE * FROM tstct
WHERE sprsl EQ sy-langu
AND tcode EQ p_tcode.

FORMAT COLOR COL_POSITIVE INTENSIFIED OFF.
WRITE:/(19) 'Transaction Code - ',
20(20) p_tcode,
45(50) tstct-ttext.
SKIP.
IF NOT jtab[] IS INITIAL.
  WRITE:/(105) sy-uline.
  FORMAT COLOR COL_HEADING INTENSIFIED ON.

*      sorting the internal table
  SORT jtab BY object.
  DATA : wf_txt(60) TYPE c,
  wf_smod TYPE i ,
  wf_badi TYPE i ,
  wf_object2(30) TYPE c.
  CLEAR : wf_smod, wf_badi , wf_object2.

*GET the total smod.
  LOOP AT jtab INTO wa_tadir.
    AT FIRST.
      FORMAT COLOR COL_HEADING INTENSIFIED ON.

      WRITE:/1 sy-vline,
      2 'Enhancement/ Business Add-in',
      41 sy-vline ,
      42 'Description',
      105 sy-vline.
      WRITE:/(105) sy-uline.
    ENDAT.
    CLEAR wf_txt.
    AT NEW object.
      IF wa_tadir-object = 'SMOD'.
        wf_object2 = 'Enhancement' .

```

```

ELSEIF wa_tadir-object = 'SXSD'.
    wf_object2 = ' Business Add-in'.

ENDIF.
FORMAT COLOR COL_GROUP INTENSIFIED ON.

WRITE:/1 sy-vline,

2 wf_object2,
105 sy-vline.
ENDAT.

CASE wa_tadir-object.
    WHEN 'SMOD'.
        wf_smod = wf_smod + 1.
        SELECT SINGLE modtext INTO wf_txt
        FROM modsapt
        WHERE sprsl = sy-langu
        AND name = wa_tadir-obj_name.
        FORMAT COLOR COL_NORMAL INTENSIFIED OFF.

    WHEN 'SXSD'.
*   for badis
        wf_badi = wf_badi + 1 .
        SELECT SINGLE text INTO wf_txt
        FROM sxs_attrt
        WHERE sprsl = sy-langu
        AND exit_name = wa_tadir-obj_name.
        FORMAT COLOR COL_NORMAL INTENSIFIED ON.
ENDCASE.

WRITE:/1 sy-vline,
2 wa_tadir-obj_name HOTSPOT ON,
41 sy-vline ,
42 wf_txt,
105 sy-vline.
AT END OF object.
    WRITE : /(105) sy-uline.
ENDAT.
ENDLOOP.

WRITE:/(105) sy-uline.

SKIP.
FORMAT COLOR COL_TOTAL INTENSIFIED ON.
WRITE:/ 'No.of Exits:' , wf_smod.
WRITE:/ 'No.of BADIs:' , wf_badi.

ELSE.
    FORMAT COLOR COL_NEGATIVE INTENSIFIED ON.
    WRITE:/(105) 'No userexits or BADIs exist'.
ENDIF.
ELSE.
    FORMAT COLOR COL_NEGATIVE INTENSIFIED ON.
    WRITE:/(105) 'Transaction does not exist'.
ENDIF.

AT LINE-SELECTION.

DATA : wf_object TYPE tadir-object.
CLEAR wf_object.

GET CURSOR FIELD field1.
CHECK field1(8) EQ 'WA_TADIR'.
READ TABLE jtab WITH KEY obj_name = sy-lisel+1(20).
MOVE jtab-object TO wf_object.

```

```
CASE wf_object.
  WHEN 'SMOD'.
    SET PARAMETER ID 'MON' FIELD sy-lisel+1(10).

    CALL TRANSACTION 'SMOD' AND SKIP FIRST SCREEN.
  WHEN 'SXSD'.
    SET PARAMETER ID 'EXN' FIELD sy-lisel+1(20).
    CALL TRANSACTION 'SE18' AND SKIP FIRST SCREEN.
ENDCASE.
```

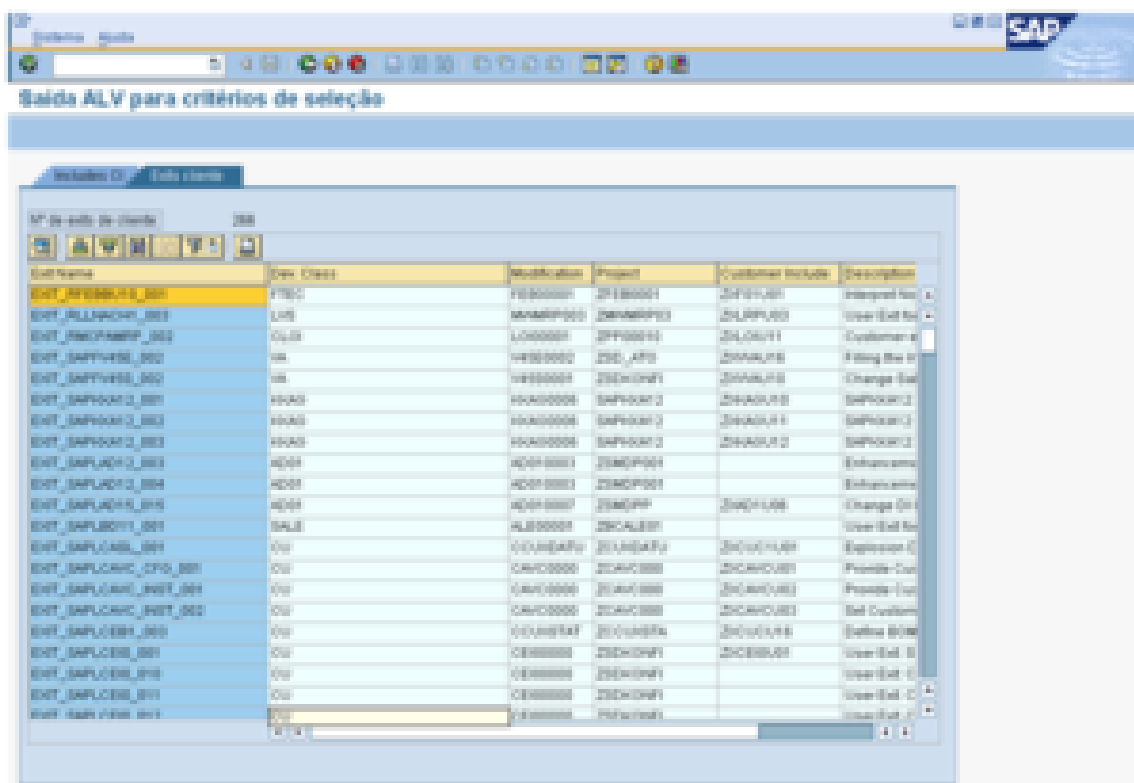
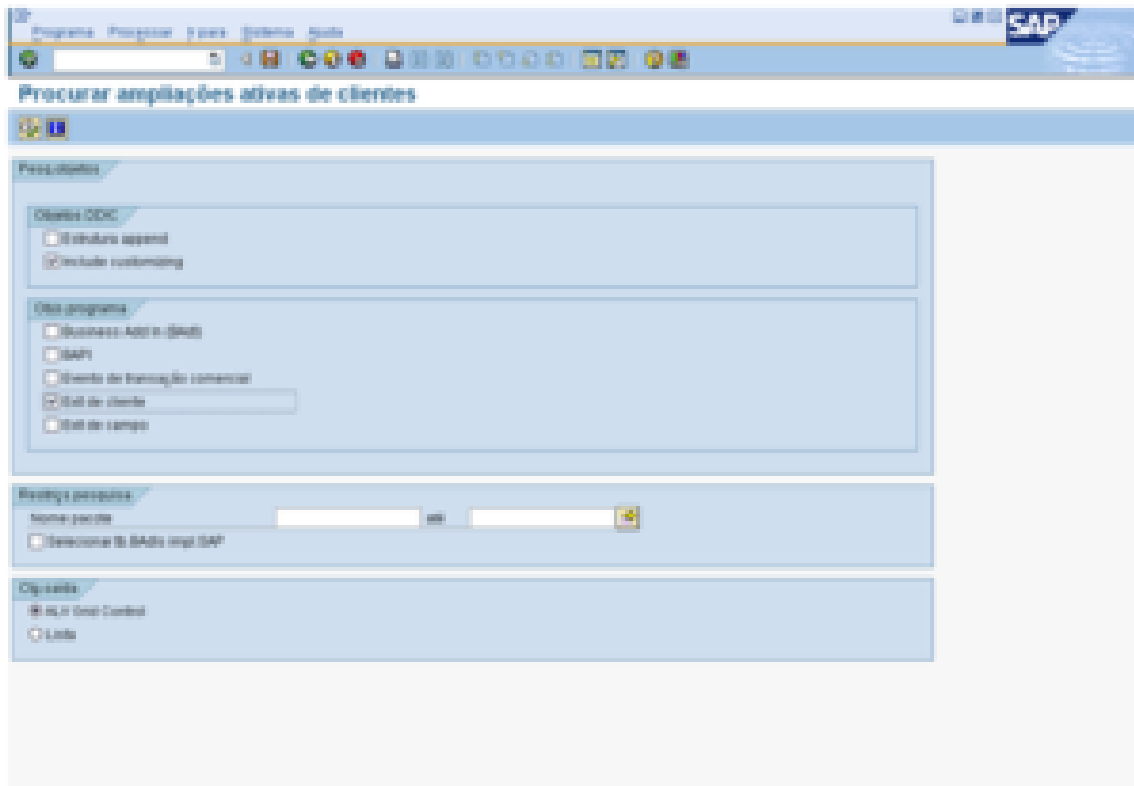
Encontrei também esse outro relatório abaixo. O código-fonte está [aqui](#). Basicamente, tem a mesma funcionalidade, mas eu particularmente prefiro o anterior.

[illegible]

Outras ferramentas

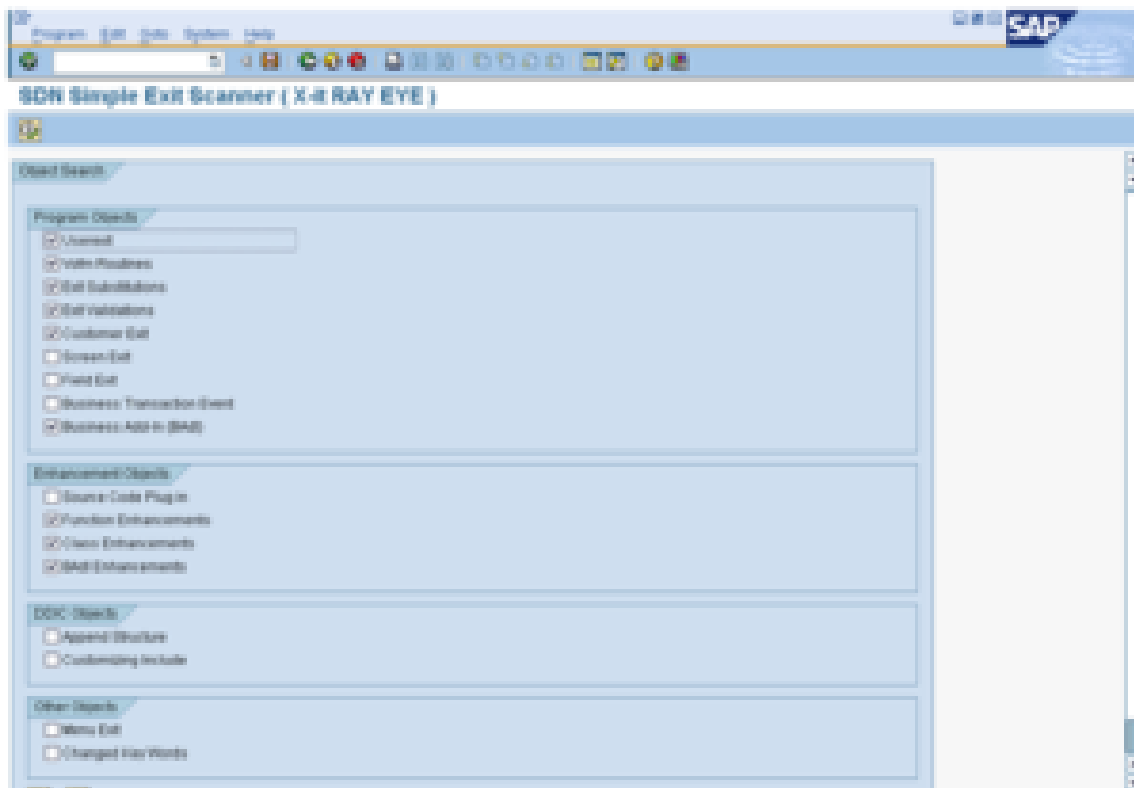
SNIF

E você pode estar se perguntando: “não existe nenhum relatório standard?”. Sim, existe! A SAP disponibiliza o SNIF (esse é o nome do programa, não da transação) que lista todos os Costumer Enhancements ativos:



ABAP X-RAY EYE

Pesquisando um pouco mais encontrei um "SNIF melhorado", que inclui algumas opções, como por exemplo, exits para Validação e Substituição, fórmulas (VOFM), Screen-exits e Field-exits. Esse projeto está disponível neste link: <http://code.google.com/p/abapextrayeye/> e para implementá-lo você precisará usar o [SAP Link](#).



The screenshot shows the SAP SDN Single Exit Scanner (X-RAY EYE) interface with the 'User Exit' tab selected. The table displays a list of user exits with the following columns: Routine Name, Include Name, Description Text, Date-Class, Created By, and Created On.

Routine Name	Include Name	Description Text	Date-Class	Created By	Created On
Form01 (1)	Form01 (1)		VB00	SAP	05.09.1999
GET_QUANTITY_FOR_RELEASE	VB01 (1) (1)	User Exit From SAP/ABAP - Release - Calculating Opp	VB00	SAP	05.09.1999
GET_QUANTITY_FOR_UPDATE	VB01 (1) (1)		VB00	SAP	05.09.1999
MONI_USERFIELD	VB01 (1) (1)	Customer enhancements for additional display field	VB00	SAP	05.09.1999
VB01 (1) (1)	VB01 (1) (1)		VB00	SAP	05.09.1999
VB01 (1) (1)	VB01 (1) (1)		VB00	SAP	05.09.1999
VB01 (1) (1)	VB01 (1) (1)		VB00	SAP	05.09.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit for BADI Determination	VB00	SAP	05.12.1999
VB01 (1) (1)	VB01 (1) (1)		VB00	SAP	05.04.1999
VB01 (1) (1)	VB01 (1) (1)	Billing doc. user exits for partner functions (1)	VB00	SAP	22.07.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit	VB	SAP	05.12.1999
VB01 (1) (1)	VB01 (1) (1)		VB00	SAP	10.07.1999
VB01 (1) (1)	VB01 (1) (1)		VB00	SAP	10.07.1999
VB01 (1) (1)	VB01 (1) (1)		VB	SAP	05.12.1999
VB01 (1) (1)	VB01 (1) (1)	Pricing Modifications	VB00	SAP	04.04.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit	VB00	SAP	04.04.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit	VB00	SAP	04.04.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit	VB00	SAP	04.04.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit	VB00	SAP	04.04.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit	VB00	SAP	04.04.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit	VB00	SAP	04.04.1999
VB01 (1) (1)	VB01 (1) (1)	User Exit	VB00	SAP	04.04.1999

Links Úteis:

[WIKI: Enhancements and Modifications – BADI, Enhancement Framework, User Exits, BTE](#)

[WIKI: Ways to find a userexit](#)

Comentários

Rafael Martin — 24/06/2022 10:32

Excelente POST, muito obrigado

Fernando Castro — 01/06/2021 12:23

Ótimo post, parabéns!

Fernando Castro — 11/11/2020 16:36

Ótimo post, parabéns.

Diego Henrique da Silva — 24/07/2020 16:07

Criei um ponto de ampliação ZTM_REFERENCIA_TRANSP na BADI (BADI_SD_BILLING), agora preciso criar uma implementação, quando clico para criar ele pede pra digitar uma definição BADI e não sei oq digitar neste campo, podem me ajudar ?

Renan — 11/07/2016 08:22

Vlw Daiane, vou trabalhar com BADI's pela primeira vez hoje, e não tenho nenhuma ideia de qual BADI devo usar para transferência de material, como descubro uma BADI que servirá para mim, sem saber onde ela já foi usada ? oh god!

Estevam Vieira — 13/12/2013 08:34

Bom dia, Daiane!

Parabéns pela iniciativa, seu site é demais, e uma ótima ferramenta.

Gostaria de saber se você poderia me indicar dois ABAPs, eu preciso de Trainee/Jr para região do Morumbi e Pleno para região da Av. Paulista. Meu email é rh@vitroit.com.br

Muito Obrigado!

Robson — 25/11/2012 14:25

Muito bacana o post!

Só um detalhe: o link "clique aqui" do item "CALL CUSTOMER FUNCTION" está quebrado.

Abs!

Mauricio Cruz — 28/11/2012 10:48

Opa, corrigido. Valeu Robson!

Andréa — 11/11/2012 09:12

Aii que orgulho, DAIII!

Nunca tinha lido um post da sua autoria, mujer! Congrats!

Adorei 😊

E me ajudou muito a lembrar... Ai ai, dá uma saudade e um aperto no meu coração ver código ABAP.. Mais do que eu sinto saudade da minha família. E eu gostaria de estar brincando hahahaha 😊

Vinicius — 20/09/2012 11:35

Meu método favorito é buscando no SDN! :p

Outra forma também é usando a boa e velha SE84 -> Enhancements -> Business Add-ins -> Definitions -> e aí busco pelo pacote da transação (ME, por exemplo). O mesmo serve pra customer-exits 😊

Danilo Cardoso — 05/09/2012 19:10

Fala galera!

Esse post ficou muito loko Daiane!

Eu tbm não manjava esses relatórios não. Curto muito aprender essas paradas, obrigado pelas dicas.

Estava em conversa com a nossa amiga do coração, Priscila Silva. Falávamos justamente sobre como encontrar BAdIs no sistema e eu escrevi um e-mail para ela detalhando dois métodos. Mas não postarei o e-mail aqui, pois, além de ser uma informação redundante, eu escrevi em inglês (só de brincadeirainha..rs) e utilizei muitas gírias do Bronx ... algumas pessoas podem achar meio nocivo (mas a Pri riu pra caramba .. rs).

Abraços.

Mauricio Cruz — 05/09/2012 22:53

SERIOUSLY, ARE YOU KIDDING ME !?! PUT THAT MOTHERFUCKING BREAK POINT IN THE MOTHERFUCKING CALL-CUSTOMER FUNCTION INSTRUCTION OR I'LL SHOOT YOU IN YOUR GODDAMN FACE!

(foi also assim? heoaueoahe)

Mauricio Cruz — 23/07/2012 10:16

Animal o post Daiane, eu não manjava desses relatórios não... eu sou meio old-school, sempre tento achar exits c/ os break-points em funções e etc! haha. Mas talvez eu tenha que mudar um pouco, esses relatórios que você postou me parecem muito úteis!

Viva o Open Source, e valeu pelo post!

Renato Bonini — 20/07/2012 16:23

Só um pitaco para achar BADIs, standards...

Por "padrão" as Classes Standard da SAP para BADIs tem a nomenclatura "CL_EX_xxxxxxx". Desta forma, acessando a transação SE24 e buscar por "CL_EX_*", no campo nome da classe. Vai listar quase todas as BADIs... e caso você não consiga chegar na BADI em si pela SE24. Use o complemento do nome da classe na SE18 ou SE19, exemplo:

classe: CL_EX_AC_DOCUMENT => BADI: AC_DOCUMENT

[]s