



Extrait du Environnement iSeries

<https://xdocs400.com/spip.php?article258>

Capture d'écran en RPGLE

- Les articles -



Date de mise en ligne : jeudi 24 novembre 2005

Description :

BUFSCRSVR permet de capturer un écran AS400 (tout ou une partie). De le stocker puis de l'éditer.

Environnement iSeries

Ce programme de service (BUFSCRSRV sqlrpgle) permet d'effectuer des captures écrans sur AS400 et de les éditer.

Il contient 3 procédures exportées :

- RtvSrc pour capturer l'écran (ou une partie de l'écran).
- BufIns qui inscrit le buffer dans la base.
- BufPrt qui permet d'éditer la capture.

Exemple d'utilisation

En général il faut enchaîner RtvScr et BufIns dans un programme interactif pour capturer et enregistrer la capture dans la base (dans le fichier BUFSCR). La clef de la capture (un TIMESTAMP associé à un profil utilisateur) est ensuite transmise au programme (par paramètres) chargé de l'édition. Il suffit d'invoquer BufPrt dans le programme d'édition pour imprimer la capture (le prtf est interne au pgm de service).

➤ Exemple de programme qui effectue la capture, l'inscription et l'édition en utilisant BUFSCRSRV. Il reçoit en paramètre un caractère qui correspond au type de l'écran ('1'=>80 colonnes, '2'=>132 colonnes).

```
H copyright('Serge GOMES 2005') H Option( *SrcStmt : *NoDebugIO ) *----- * Récup du buffer de l'écran encour * Ce
PGM utilise le programme de service BUFSCRSRV *----- * Compilation => CrtRPGMod Module(YourLib/CAPSCR) *
SrcFile(YourLib/QRPGLESRC) * CrtPgm Pgm(YourLib/CAPSCR) * BndSrvPgm(YourLib/BUFSCRSRV) * ActGrp(*New)
*----- * Paramètres : * TypEcran 1A '1'=> 80*24 '2'=> 132*27
*----- /COPY SERGE/PROTOTYPE,BUFSCRSRV DP_type S 1A DW_coldeb s 2 0 DW_cofin
s 2 0 DBufferEcran S 3565A DW_time S z DW_supbuf s n inz(*on) DW_cadre s n inz(*on) C *Entry Plist
C Parm P_type *aCapture Buffer C Eval BufferEcran=RtvScr( C P_type) *aInscription Buffer C If
BufferEcran <> *blank C Eval W_time=BUFINS(P_type:BufferEcran) *aImpression buffer C Callp BUFPRT(W_time) C Endlf C
Eval *InLR = *On
```

Pour effectuer une capture écran et la éditer écrivons `CALLP BUFPRT(BUFINS(P_TYPE:RTVSCR(P_TYPE)))`

➤ Compilation de l'exemple

```
CRTPGM PGM(BIB/CAPSCR) BNDSRVPGM(BIB/BUFSCRSRV) ACTGRP(*NEW)
```

Installation du programme de service

➤ Source du liage (BIB/QSRVSR)

Si le fichier source QSRVSR n'existe pas il faut le créer.

```
STRPGMEXP PGMLVL(*CURRENT) SIGNATURE('BUFSCRSRV_V01')
EXPORT SYMBOL("RTVSCR")
EXPORT SYMBOL("BUFPRT")
EXPORT SYMBOL("BUFINS")
ENDPGMEXP
```

➤ Création du fichier BUFSCR

(remplacer SERGE par votre bib)

Capture d'écran en RPGLE

```
CREATE TABLE SERGE/BUFSCR
(USRSCR CHAR ( 10) NOT NULL WITH DEFAULT,
TITSCR CHAR (130 ) NOT NULL WITH DEFAULT,
TIMSCR TIMESTAMP NOT NULL WITH DEFAULT,
TYPSCR CHAR(1) NOT NULL WITH DEFAULT,
BUFSCR CHAR (3565 ) NOT NULL WITH DEFAULT)

CREATE UNIQUE INDEX SERGE/BUFSCR1 ON SERGE/BUFSCR
(USRSCR, TIMSCR)
```

➤ Création du programme de service

— Téléchargez le code source bufsrcsrv.txt

— Copiez bufsrcsrv.txt dans un membre de type sqlrpgle de votre fichier source (qrpgrsrc).

— Copiez les prototypes ci-dessous et placez les dans votre fichier source.

PROTOTYPE

```
*-----* * CAPTURE DU BUFFER ECRAN CONTENU RTVSCR * *-----* *
Récup du buffer de l'écran encours RTVSCR * Valeur de renvoi => 3565A (buffer Ecran) * Paramètres => * 1A Type Ecran '1'=80 colonnes '2'=132colonnes
OBLIGATOIRE * 2P0 Ligne début capture OPTIONNEL * 2P0 Ligne fin capture OPTIONNEL * 272A Erreur
OPTIONNEL *-----* * - RtvScr Capture écran //IF DEFINED(BUFSRCSRVR) /EOF /ENDIF /DEFINE BUFSRCSRVR D
RtvScr PR 3565A D 1A D 2 0 Options(*NoPass) D 2 0 Options(*NoPass) D 272
Options(*NoPass)

*-----* * EDITION DU BUFFER ECRAN CONTENU BUFPRT * *-----* *
PARAMETRES : * * 1 - TIMESTAMP capture 26 Z * * 2 - Colonne de début de la capture 2 ;0 * * 3 -
Colonne de fin de la capture 2 ;0 * * 4 - Editer cadre capture N * * 5 - Supprimer le buffer d'écran dans BASBUF O/N N * * 6 -
User de la capture 10 A * *-----* * D BufPrt PR D P_time z D P_coldeb 2 0
options(*NoPass) D P_colfin 2 0 options(*NoPass) D P_cadre n options(*NoPass) D P_supbuf n options(*NoPass) DP_user
10a options(*NoPass)

*-----* * Enregistrement du buffer écran dans la BUFSCR * *-----* *
de renvoi => TIMESTAMP constituant la clef d'accès * Paramètres => * 1A Type Ecran '1'=80 colonnes '2'=132colonnes OBLIGATOIRE * 3565A Buffer écran
OBLIGATOIRE * 136A Titre capture OPTIONNEL * 10A User capture OPTIONNEL
*-----* * D BufInz PR z D P_type 1A D P_buffer 3565A D P_titre 136A Options(
*NoPass) D P_user 10A Options(*NoPass)
```

— Compilation

* Création du module

```
CRTSQLRPGI ??OBJ(BIB/BUFSRCSRVR)
```

```
?*SRCFILE(BIB/QRPGLESRC)
```

```
?*SRCMBR(BUFSRCSRVR)
```

```
?*OBJTYPE(*MODULE)
```

```
??REPLACE(*YES)
```

* Création SRVPGM

```
CRTSRVPGM SRVPGM(SERGE/BUFSRCSRVR)
```

```
SRCFILE(QSRVSRV)
```

```
ACTGRP(*CALLER)
```

Descriptions des procédures de BUFSRCSRVR

➤ RtvScr permet de capturer l'écran.

— Cette procédure renvoie une chaîne de 3565 caractères contenant la capture.

— Le 1er paramètre, obligatoire correspond au type d'écran candidat à la capture. '1' = 80 col et '2' = 132 col.

— Les 2èmes et 3èmes paramètres optionnels (par défaut tout l'écran est capturé) permettent d'indiquer une ligne de début et une ligne de fin pour la capture, le reste de l'écran sera ignoré.

— Le 4ème paramètre concerne la gestion des erreurs.

➤ BufIns permet d'inscrire le buffer dans la base (fichier BUFSCR).

— Le buffer capturé peut être stocké (temporairement) dans le fichier BUFSRC. La procédure renvoie un TIMESTAMP (Ing 26) si l'inscription à réussie.

— 1er paramètre : Type d'écran ('1' ou '2')

— 2ème paramètre : Ecran capturé (chaîne de 3565A)

— Le 3ème paramètre est optionnel. Vous pouvez indiquer un titre pour votre capture(136A). Ce titre apparaîtra (si il est renseigné et <> *blank) sur l'état.

— Le 4ème paramètre vous permet d'indiquer un user pour la capture (par défaut c'est l'utilisateur du job en cours qui est utilisé).

➤ BufPrt permet d'éditer la capture et ne renvoie aucune valeur.

— Le 1er paramètre, obligatoire est le TIMESTAMP récupéré par BufIns.

— Les 2èmes et 3èmes paramètres optionnels (par défaut tout l'écran est capturé) permettent d'indiquer une colonne de début et une colonne de fin pour la capture, le reste de l'écran sera ignoré.

— Le 4ème paramètre permet de tracer un cadre autour de la capture (*on=cadre *off=pas de cadre). Par défaut il y a un cadre.

— Le 5ème paramètre permet d'indiquer si on souhaite supprimer le buffer du fichier BUFSCR

(*on=suppression *off=pas de suppression). Cela permet d'utiliser la capture pour plusieurs impressions. Si ce paramètre n'est pas renseigné le buffer est supprimé à la fin de l'édition.

— Le 6ème paramètre correspond à l'user de la capture. Si vous avez indiqué un user spécifique lors de l'inscription du buffer, vous devez renseigner ce paramètre avec le même user. Par défaut l'utilisateur du job est utilisé.

Téléchargement



BUFSCRSRV Source du programme de service

Il existe aussi une version ADELIA de capture d'écran [Voir la version ADELIA](#)

Post-scriptum :

Pensez à corriger les directives /copy dans vos sources en indiquant le chemin de vos prototypes.