



Extrait du Environnement iSeries

<http://xdocs400.com/spip.php?article337>

# La gestion des messages sur ISeries

- Les articles -



Date de mise en ligne : jeudi 3 mai 2007

## **Description :**

Présentation condensée du système de message sur ISeries

---

**Environnement iSeries**

---

## INTRODUCTION

Un message est un moyen de communiquer entre :

- des utilisateurs
- des programmes
- un programme et un(des) utilisateur(s).

On distingue deux types de messages :

- les messages « instantanés » :

Créés temporairement par un programme ou par un utilisateur. Ces messages ne sont pas conservés. Le texte est écrit lors de l'envoi du message. Dans ce type de message on envoie une chaîne de caractères (littérale ou encapsulé dans une variable).

- Les messages « prédéfinis » :

Créés et stockés avant leurs utilisations. Ces messages sont stockés dans des fichiers de messages (objet AS/400 de type \*MSGF). Cela permet de modifier des messages sans modifier les programmes (appréciable pour le support multi-langages). L'OS/400 comporte de très nombreux messages prédéfinis, qui permettent la communication au sein du système, et entre le système et les utilisateurs.

Pour envoyer un message de ce type, il faut indiquer son identifiant et le nom du fichier messages.

## Description de messages (MSGD)

Un message prédéfini correspond à une description de message (MSGD : Message Description) appartenant à un fichier message (MSGF). Une description de message contient un identifiant (7 caractères) unique dans le fichier de messages. La commande ADDMSGD (Add Message Description) permet d'ajouter un message dans un fichier de messages (MSGF) et renseigner

- le texte du message (MSGID) - le texte d'aide (MSG) - le niveau de gravité de ce message (SECLVL)

## Les files d'attentes de messages

Tous les messages envoyés ou reçus sont transmis à travers les files d'attente de messages (MSGQ). Il y a plusieurs files d'attente de messages fournies avec l'OS/400. Le système envoie également des informations d'état du système, appelé « log », concernant les travaux, les sous-systèmes et les unités.

Il existe plusieurs types de files d'attente de messages.

➤ MSGQ associée à un écran.

&mdash; Elles ont le même nom que l'écran &mdash; Créées implicitement à la première utilisation de l'écran  
&mdash; Allouées à l'écran de même nom &mdash; Suppression impossible

➤ MSGQ utilisateur :

Créées par les utilisateurs (ou à la création du profil) pour recevoir les messages dans une MSGQ particulière.

➤ MSGQ du travail :

Elles sont créées automatiquement par le système pour chaque travail et sont constituées de 2 parties :

&mdash; la MSGQ externe (\*EXT) Elle sert à communiquer entre le travail et l'utilisateur pour les travaux interactifs. Tous les messages reçus sont dupliqués dans l'historique du travail (JOBLOG), cette JOBLOG peut être visualisée ou imprimée grâce à la commande DSPJOB option 10 ou par DSPJOBLOG. Pour un travail interactif si un message est envoyé à la MSGQ externe le message est visualisé à l'écran et le travail attend une réponse pour les messages d'interrogation ou d'avertissement. Si un message d'état du travail est envoyé il sera visualisé sur la ligne de message de l'écran. Pour un travail BATCH. Le système affectera automatiquement la réponse par défaut si des messages d'interrogation ou d'avertissement sont envoyés. Cette partie reçoit toutes les demandes d'exécution de commandes. Quand vous appelez un programme un message est envoyé à la MSGQ externe de type demande (request RQS) ce message est traité par le système pour exécuter la commande. Quand vous soumettez un travail vous écrivez la commande à exécuter dans le paramètre RQSDTA (donnée de la demande).

&mdash; les MSGQ programme ce type de MSGQ est créée pour chaque invocation d'un programme qui envoie des messages ou en reçoit, cette MSGQ aura le même nom que le programme.

➤ Il existe enfin une file d'attente de messages particulière, dans laquelle sont envoyés et stockés les « logs » du système. Cette file d'attente s'appelle QHST. On y accède par la commande DSPLOG (afficher historique du système).

Vous pouvez visualiser le contenu d'une MSGQ par la commande DSPMSG.

`DSPMSG MSGQ(nom MSGQ)`

➤ Mode de livraison d'une MSGQ :

Les MSGQ peuvent être allouées par un utilisateur grâce à la commande CHGMSGQ, cette commande sert aussi à changer le mode de livraison des messages.

`CHGMSGQ MSGQ(nom de la msgq)`

`DLVRY(mode)`

4 modes de réception :

&mdash; \*BREAK : Le travail qui s'est alloué la MSGQ est interrompu.

&mdash; \*HOLD : Les messages sont stockés dans la MSGQ, ils sont en attente (mode par défaut pour les MSGQ utilisateurs).

&mdash; \*NOTIFY : Les messages sont stockés et en attente. Lorsqu'un message arrive si c'est un travail interactif qui s'est alloué la MSGQ, l'indicateur d'erreur s'allume (mode par défaut pour les MSGQ écrans).

&mdash; \*DFT : Les messages demandant une réponse reçoivent la réponse par défaut qui leur correspond. Aucun message n'est ajouté à la file d'attente de messages, excepté s'il s'agit de QSYSOPR.

### ➤ Modification de MSGQ (exemple)

Le mode \*BREAK pour une MSGQ utilisateur ou écran est un peu gênant car le message s'affiche sur tout l'écran. On peut modifier ces MSGQ grâce à un Programme gestion interruption (ici BRKMSG) pour pouvoir recevoir les messages d'interruption en bas de l'écran (status)

Il faut changer les MSGQ

&mdash; CHGMSGQ MSGQ(\*USRPRF) PGM(BRKMSG)

&mdash; CHGMSGQ MSGQ(\*WRKSTN) PGM(BRKMSG)

&mdash; Code Source BRKMSG

```
PGM          PARM(&msgq &msgqlib &msgkey)
DCL          VAR(&msgq) TYPE(*CHAR) LEN(10)
DCL          VAR(&msgqlib) TYPE(*CHAR) LEN(10)
DCL          VAR(&msgkey) TYPE(*CHAR) LEN(4)
DCL          VAR(&msgdta) TYPE(*CHAR) LEN(132)
RCVMSG      MSGQ(&msgqlib/&msgq)
MSGKEY(&msgkey)
RMV(*NO) MSG(&msgdta)
SNDPGMMSG   MSGID(CPF9897) MSGF(QCPFMSG)
MSGDTA(&msgdta) TOPGMQ(*EXT)
MSGTYPE(*STATUS)
ENDPGM
```

### ENVOI DE MESSAGES.

Pour envoyer un message il faut au moins indiquer, le texte du message, son type et la MSGQ destinatrice.

Les messages de type \*INFO peuvent être envoyés à plusieurs msgq, par contre ceux \*INQ ne peuvent être envoyés qu'à une seule msgq. Les autres type de messages sont utilisables uniquement avec la commande SNDMSGPGM.

\*INFO (information) message d'information \*INQ (interrogation) question nécessitant une réponse \*COMP (achèvement) indication de fin de travail \*DIAG ( diagnostic) indication d'erreur pendant l'exécution du travail \*RQS (demande) demande d'exécution d'une fonction \*ESCAPE (abandon ) indication de fin anormale \*STATUS ( état) état d'avancement d'un travail \*NOTIFY(avertissement) demande d'intervention pour continuer le travail

Il existe 4 commandes pour envoyer des messages :

### ➤ SNDMSG

Pour envoyer uniquement des messages instantanés (pas de MSGF) à des msgq écrans, utilisateurs ou de

l'opérateur QSYSOPR.

```
SNDDMSG MSG(texte du message)
TOMSGQ(nom de la msgq destinatrice)
MSGTYPE(type du message *INFO ou *INQ)
RPYMSGQ(nom de la msgq qui recevra la réponse d'1 message *INQ, par défaut celle de l'écran)
```

### ➤ SNDBRKMSG

Pour envoyer uniquement des messages instantanés (pas de MSGF) à des msgq écrans. L'envoi d'un message par cette commande provoque l'arrêt du travail en cours et affiche le message (même comportement qu'un SNDDMSG vers une MSGQ écran en mode réception \*BREAK)

```
SNDBRKMSG MSG(texte du message)
TOMSGQ(nom de la msgq destinatrice, par défaut *ALLWS tous les écrans)
MSGTYPE(*INFO ou *INQ)
RPYMSGQ(nom de la msgq qui recevra la réponse d'1 message *INQ, par défaut QSYSOPR)
```

### ➤ SNDUSRMSG

Permet d'envoyer des messages prédéfinis ou instantanés à la MSGQ externe d'un travail, à QSYSOPR ou à une MSGQ écrans ou utilisateurs. Pour les messages d'interrogation (MSGTYPE=\*INQ), on peut récupérer la réponse dans une variable (132 caractères maximum).

Par exemple pour envoyer un message prédéfinis (MSG001 de ERMSGF) à l'écran et récupérer la réponse dans la variable &REP

```
SNDUSRMSG MSGID(MSG001) MSGF(ERMSGF) MSGTYPE(*INQ) TOMSGQ(*) MSGRPY(&REP)
IF COND(&REP *EQ 'N') THEN(GOTO CMDLBL(TRT_N))
```

### ➤ SNDPGMMSG

Permet d'envoyer des messages prédéfinis ou instantanés à des MSGQ écrans, programmes ou utilisateurs. Cette commande est beaucoup utilisée pour la communication entre programmes.

```
SNDPGMMSG MSG(texte du message si message impromptu)
MSGID(clé du message si message prédéfini)
MSGF(nom du fichier des messages si msg prédéfini)
MSGDTA(si le message prédéfini contient des variables vous indiquerez ici la valeur des variables)
TOPGMQ(nom de la file d'attente programme ).
TOMSGQ(nom d'une file d'attente de messages écrans ou utilisateurs, incompatible si TOPGMQ renseigné)
MSGTYPE(9 types de message possibles)
RPYMSGQ(pour les messages *INQ ou *NOTIFY le nom de la pgmq ou de la msgq qui recevra la réponse)
KEYVAR(nom d'une variable qui récupérera le numéro identifiant du message, utile pour relire le message)
```

*Post-scriptum :*

GO CMDMSG\* pour accéder à tous les menus concernant les messages.