

TBT/400 et l'utilisation de la Haute disponibilité

| | |
|--|---|
| 1 Remarques initiales..... | 2 |
| 1.1 Journalisation..... | 2 |
| 1.2 Audit..... | 2 |
| 1.3 Vacation par comparaison ou forcée..... | 2 |
| 1.4 Quelles méthodes choisir pour les objets de TBT/400 ?..... | 3 |
| 2 Les objets de TBT/400..... | 4 |
| 2.1 Les Bibliothèques..... | 4 |
| 2.1.1 IPLSP..... | 6 |
| 2.1.2 IPLSC..... | 6 |
| 2.1.3 IPLSE IPLSM UserE..... | 7 |
| 2.1.4 IPLSEnn UserEnn et IPLSMnn (Version 610M68)..... | 7 |
| 2.1.5 QSYS..... | 8 |
| 2.2 Répertoires utilisés par TBT/400..... | 8 |
| 2.2.1 /IFZTBTIPZ..... | 8 |
| 2.2.2 /IFSTBTIPZC..... | 8 |
| 2.2.3 /IFSTBTIPZW..... | 8 |
| 3 Composants OS/400 mis à jour par TBT/400..... | 8 |
| 3.1 Commandes OS/400 de réplication manuelle de l'environnement SSL..... | 8 |

1 Remarques initiales

La réplication pour haute disponibilité peut se faire de plusieurs manières :

- par l'utilisation de la **journalisation** (possible pour les objets de type *FILE *DTAQ *DTAARA),
- par l'utilisation de l'**audit** (possible pour pratiquement tous les objets),
- à la vacation, par **comparaison** avec le système cible,
- à la vacation, de **manière forcée**.

1.1 Journalisation

La **journalisation** d'un objet écrit les informations d'ajout, création, suppression dans un journal 'mirroiré' sur le site distant ou le '**film est rejoué quasiment en temps réel**'. Elle est de fait adaptée aux objets de type bases de données (TBT/400 n'utilise pas de *DTAQ ni de *DTAARA permanente en mise à jour).

1.2 Audit

L'**audit** permet à l'outil de réplication d'**identifier les objets modifiés** (contenu, libellé, règles de sécurité, owner, ...) et de les reproduire sur le site distant de manière asynchrone.

1.3 Vacation par comparaison ou forcée

Ces deux méthodes répondent le plus souvent à un besoin ponctuel et sont pilotées par l'utilisateur.

1.4 Quelles méthodes choisir pour les objets de TBT/400 ?

Il paraît évident que le type de réplication à mettre en œuvre dépend de la vitesse de la ligne disponible ; il est logique de différencier la réplication 'locale' (utilisation d'un lien à très grand débit ≥ 100 Méga) de la réplication distante (utilisation d'un lien à 1 ou 2 Méga) :

- **Intérêt de la réplication des fichiers séquentiels** (IPLSE et IPLSM) : les fichiers **TBT/400** étant des objets de nature temporaire, il paraît souhaitable que leur réplication n'introduise pas une latence trop grande du système de réplication : Un fichier ne sera considéré comme valide que si un événement a été déposé dans la base de données **IPSHISP**, la réplication (par journalisation) de ce fichier permettra en tout état de cause, lors d'un redémarrage de la machine de backup, d'être informé d'un problème. Nous suggérons d'éviter la journalisation des fichiers séquentiels, sauf réplication 'locale' (à très grand débit) et d'utiliser l'Audit si on souhaite répliquer ces fichiers.
- D'où nos suggestions :
 - IPLSC
 - Répliquer par **journalisation** les objets *FILE (PF-DTA)
 - Répliquer par **Audit** ou à la vacation les objets autres
 - IPLSP
 - Répliquer par **Audit** ou à la vacation les objets
 - **IPLSE, UserE et IPLSM**
 - Répliquer par **Audit** les objets si la ligne le permet, à la vacation sinon
 - Il est également possible d'utiliser la journalisation (plus consommateur mais permettant une charge lissée : attention le temps d'écriture est dégradé et certains transferts peuvent y être très sensibles.),
 - **IPLSEnn, UserEnn et IPLSMnn**
 - **TBT/400** propage l'attribut **CRTOBJAUD** de **IPLSE IPLSM** et **UserE**; si le choix **Audit** a été retenu au point c) il s'applique naturellement aux bibliothèques quotidiennes.
 - Quant à la journalisation,
 - **TBT/400** n'apporte actuellement aucun support quant à la reconduction de la journalisation de IPLSE IPLSM UserE. Jusqu'à la V6R1M0 de l'OS/400 exclus
 - A partir de la V6R1M0 de l'OS/400, si une journalisation est en cours sur une des bibliothèques IPLSE IPLSM USERE, elle sera reconduite sur les bibliothèques datées correspondantes.
 - *JOBQ *MSGQ
 - Il n'est pas obligatoire de répliquer leur contenu (rien de critique)
 - *OUTQ
 - Quant au contenu des *OUTQ, TBT/400 ne génère que des spools de type trace dont le contenu n'est pas critique (on peut se dispenser de la réplication); mais, les applicatifs de traitement des fichiers reçus tournent souvent dans le sous-système TBT/400, et sont rarement customisés pour utiliser leur propre Output Queue ; en conséquence il peut exister dans les Output Queues de TBT/400 des spools créés par des applicatifs client dont on ne peut préjuger de la sensibilité (ceci dit rarement critique).
 - Répliquer par **Audit les objets IFS des trois répertoires utilisés par TBT/400**
 - Pour la **gestion des certificats** se pose la question des composants IBM :
 - Le fichier des certificats peut être répliqué sur Audit : vacation obligatoire (/QIBM/USERDATA/ICSS)
 - La table des applications pose problème ; elle est contenue dans un objet contenant tous les EXITS enregistrés du système OS/400 (elle n'est pas limitée aux applications) Ceci étant, TBT/400 la renseigne à chaque démarrage ; la réplication ne s'impose pas (QUSRSYS/QUSEXRGOBJ, de type « *EXITRG ».
 - L'association certificats / applications est contenu dans un « User Index » à priori non auditable : à la vacation (QUSRSYS/QYCDCERTI de type « *USRIDX »).

2 Les objets de TBT/400

2.1 Les Bibliothèques

| Nom | Contenu | Commentaires | Disponibilité | Suggestion de réplication |
|---------|-------------------|---|---------------|--|
| IPLSP | Progiciel | Contient les divers objets du progiciel (Programmes, Commandes, ...); elle n'évolue que lors des changements de version | | Audit |
| IPLSC | Configuration | Contient les éléments de configuration : définition de sous-système, bases de données TBT/400 | | Audit + Journalisation des *FILE |
| IPLSE | Fichiers reçus | | | Audit ou Journalisation |
| UserE | Fichiers reçus | Version 'Utilisateur' de IPLSE ; plusieurs possibles | | Audit ou Journalisation |
| IPLSM | Fichiers émis | | | Audit ou Journalisation |
| QSYS | Objets 'système', | TBT/400 peut créer dynamiquement des bibliothèques, et gérer des listes d'autorisation (*AUTL) | | Audit des *AUTL |
| IPLSEnn | Fichiers reçus | Optionnel selon paramétrage ; idem IPLSE mais une par jour. Pour très gros volumes, TBT/400 reconduisant les attributs de sécurité et | V610M68 | Audit ou Journalisation (Attention à la propagation) |

| | | | | |
|---------|----------------|--|---------|---|
| | | d'Audit de IPLSE A partir de la V6R1M0 de l'OS/400 la journalisation de la bibliothèque IPLSE sera reconduite | | |
| UserEnn | Fichiers reçus | Version utilisateur de IPLSEnn ; plusieurs jeux possibles. Comportement identique à IPLSE, la bibliothèque 'modèle' étant UserE | V610M68 | Audit ou Journalisation(Attention à la propagation) |
| IPLSMnn | Fichiers émis | Optionnel selon paramétrage ; idem IPLSM mais une par jour. Comportement identique à IPLSM, la bibliothèque 'modèle' étant UserM | V610M68 | Audit ou Journalisation(Attention à la propagation) |

L'utilisation du paramètre **CRTOBJAUD** (Audit des objets créés) permet de définir la valeur d'audit de tous les objets créés.

2.1.1 IPLSP

Cette bibliothèque est de nature 'statique' ; nous suggérons de la répliquer par l'utilisation de l'audit (cas numéro 2) ou à la vacation (cas numéros 3 et 4).

2.1.2 IPLSC

Cette bibliothèque contient les bases de données de TBT/400, la description de sous-système , ainsi qu'un Userspace (Objet de type *USRSPC).

- Pour les bases de données (objets IPS* de type FILE attribut PF) nous suggérons d'utiliser la journalisation,
- Pour les fichiers logiques (objets IPS* de type FILE attribut LF) idem (à préciser),
- Pour tous les autres objets (à l'exception du Userspace) utilisation de l'Audit (cas numéro 2) :
 - Les objets concernés 'bougent' très peu (Jobs descriptions),
 - Lors d'une réinstallation ces composants sont remplacés; l'utilisation du paramètre CRTOBJAUD au niveau de la bibliothèque propagera l'Audit sur les nouveaux objets.
- Le Userspace pose un problème : il y a deux manières de mettre à jour un Userspace : (Utilisation de l'API QUSCHGUS ou adressage et mise à jour mémoire). Seule l'utilisation de l'API QUSCHGUS peut générer un poste d'Audit.
 - Jusqu'à la version 610M67 de TBT/400, la seule manière de répliquer le Userspace est la vacation (utilisation des points 3 et 4)
 - A partir de la version 610M68, si le Userspace est en Audit, TBT/400 utilise périodiquement QUSCHGUS pour forcer un poste d'Audit rendant le cas 2 possible.

2.1.3 IPLSE IPLSM UserE

Ces bibliothèques contiennent les fichiers reçus (IPLSE, UserE) ou émis (IPLSM).

Les fichiers sont créés dynamiquement (il y a donc propagation de l'attribut CRTOBJAUD de la bibliothèque).

Ils peuvent se matérialiser, selon paramétrage, par :

- Des Savefiles,
- Des Fichiers monomembre,
- Des Fichiers multimembre globaux,
- Des Fichiers multimembre quotidiens.

Dans tous les cas nous suggérons l'utilisation de l'Audit (cas 2) ou la comparaison (cas 3).

En effet :

- Pas de journalisation pour les savefiles
- Les fichiers concernés ne sont pas des bases de données (au sens fonctionnel), mais des fichiers séquentiels. Ils ne sont exploitables que lorsque le dernier caractère a été écrit. Il paraît beaucoup moins consommateur de répliquer le composant (membre ou fichier) en une seule fois que de répliquer enregistrement par enregistrement.
- Nous recommandons dans TBT/400 de ne pas utiliser les multimembre globaux mais les multimembre quotidiens.

2.1.4 IPLSEnn UserEnn et IPLSMnn (Version 610M68)

TBT/400 permet, selon option, de créer un jeu de bibliothèques par jour ;

Dans ce cas les attributs de sécurité et d'audit des bibliothèques IPLSE sont repris pour IPLSEnn ; il en est de même pour IPLSM et IPLSMnn, UserE et UserEnn.

Nous suggérons l'utilisation de l'Audit (cas 2) ou la comparaison (cas 3) ; à noter que l'outil de réplication devra être paramétré en 'générique' (les règles de réplication concernent IPLSE* et IPLSM*).

TBT/400 propage l'attribut CRTOBJAUD des bibliothèques modèle IPLSE IPLSM et UserE.

A partir de la V6R1M0 de l'OS/400 TBT/400 reconduit également la journalisation des bibliothèques modèle.

2.1.5 QSYS

Certains objets de QSYS peuvent être modifiés (listes d'autorisation en particulier). Nous suggérons une réplification des *AUTL par audit.

2.2 Répertoires utilisés par TBT/400

2.2.1 /IFZTBTIPZ

Dans ce répertoire se crée un jeu quotidien de répertoires de deuxième niveau dans lesquels se créent les fichiers IFS émis / reçus.

TBT/400 propage l'attribut CRTOBJAUD du répertoire de 1^{er} niveau sur les autres (en standard c'est une Sysval qui est retenue).

Là aussi nous suggérons l'utilisation de l'Audit (cas 2) ou de la vacation (cas 3).

2.2.2 /IFSTBTIPZC

Ce répertoire contient les certificats locaux (*.p12) et distants (*.cer). Là aussi nous suggérons l'utilisation de l'Audit (cas 2) ou de la vacation (cas 3).

2.2.3 /IFSTBTIPZW

Ce répertoire contient le serveur Java d'administration de TBT/400 ; il ne 'bouge' que lors d'une réinstallation.

3 Composants OS/400 mis à jour par TBT/400

TBT/400 met à jour plusieurs composants de l'OS/400

- Les tables TCP/IP :
 - La table des Hosts (Mise à jour manuelle),
 - La table des ports restrictions (Recréée au démarrage du sous-système),
 - La table des services (Recréée au démarrage du sous-système),
- Le fichier des certificats de l'AS/400 (/qibm/userdata/ICSS),
- Applications dans QUSRSYS/QUSEXRGOBJ type *EXITRG,
- Association applications / certificats QUSRSYS/QYCDCERTI *USRIDX (manipulable par API).

3.1 Commandes OS/400 de réplification manuelle de l'environnement SSL

- SAVRST RMTLOCNAME(MONSERVEUR) OBJ('/qibm/userdata/icss' *INCLUDE))
- SAVRSTOBJ OBJ(QUSEXRGOBJ) LIB(QUSRSYS) RMTLOCNAME(MONSERVEUR) OBJTYPE(*EXITRG)
- SAVRSTOBJ OBJ(QYCDCERTI) LIB(QUSRSYS) RMTLOCNAME(MONSERVEUR) OBJTYPE(*USRIDX)