

Traitement transactionnel

BULL DPS 7000

Concepts et caractéristiques

TDS - Concepts

GCOS 7

Logiciel

Sujet : Ce manuel présente les concepts de TDS dans le système d'exploitation GCOS 7.

Observations :

Version du logiciel : GCOS 7-V6

Date : Novembre 1995

Bull S.A.
CEDOC
Atelier de reproduction
331, Avenue Patton BP 428
49005 ANGERS Cedex
FRANCE

Bull HN Information Systems Inc.
Publication Order Entry
MA30/843
300 Concord Road
Billerica, MA 01821
U.S.A.

Copyright © Bull S.A., 1995

Toutes les marques citées sont la propriété de leurs titulaires respectifs.

Vos suggestions sur la forme et le fond de ce manuel seront les bienvenues. Une feuille destinée à recevoir vos remarques se trouve à la fin du présent manuel.

La loi du 11 mars 1957, complétée par la loi du 3 juillet 1985, interdit les copies ou reproductions destinées à une utilisation collective. Toute représentation ou reproduction intégrale ou partielle faite par quelque procédé que ce soit, sans consentement de l'auteur ou de ses ayants cause, est illicite et constitue une contrefaçon sanctionnée par les articles 425 et suivants du code pénal.

Ce document est fourni à titre d'information seulement. Il n'engage pas la responsabilité de Bull S.A. en cas de dommages résultant de son application. Des corrections ou modifications au contenu de ce document peuvent intervenir sans préavis ; des mises à jour ultérieures les signaleront éventuellement aux destinataires.

Préface

UTILISATEURS CONCERNES

Ce manuel s'adresse à tout utilisateur intéressé par les sous-systèmes transactionnels (TDS). Il peut s'agir des utilisateurs, programmeurs, opérateurs ou administrateurs du système qui ont besoin de se familiariser avec les applications TDS ou qui s'en servent.

OBJET DU MANUEL

Ce document explique les concepts fondamentaux de TDS au sein du système d'exploitation GCOS 7. Des informations supplémentaires concernant la mise au point, l'utilisation et la gestion d'un système TDS se trouvent dans les documents suivants :

<i>TDS - Guide de l'administrateur</i>	47 F2 20UT
<i>TDS Administrator's Guide</i>	47 A2 20UT
<i>TD/C - Guide du programmeur</i>	47 F2 07UT
<i>TDS C Language Programmer's Manual</i>	47 A2 07UT
<i>TDS/COBOL - Manuel du programmeur</i>	47 F2 21UT
<i>TDS COBOL Programmer's Manual</i>	47 A2 21UT
<i>TDS- Guide pratique</i>	47 F2 04UT
<i>TDS Quick Reference Handbook</i>	47 A2 04UT
<i>Intercommunication transactionnelle sous protocole XCP1</i>	47 F2 11UT
<i>Transactional Intercommunication</i> <i>Using XCP1 Protocol User's Guide</i>	47 A2 11UT
<i>CPI-C/XCP2 - Guide de l'utilisateur</i>	47 F2 14UT
<i>CPI-C/XCP2 User's Guide</i>	47 A2 14UT

<i>Haute disponibilité (HA) - Concepts</i>	47 F2 22UT
<i>High Availability Concepts</i>	47 A2 22UT
<i>Haute disponibilité (HA) - Guide de l'utilisateur</i>	47 F2 23UT
<i>High Availability Administrator's Guide</i>	47 A2 23UT

STRUCTURE DU MANUEL

Chapitre 1	Donne un aperçu de TDS et du traitement transactionnel. Ce chapitre explique en quoi TDS diffère des autres types de traitement, et la manière dont TDS dialogue avec GCOS 7 et d'autres produits. Il présente également les différents types d'utilisateurs d'une application TDS.
Chapitre 2	Décrit les concepts fondamentaux de TDS et concerne tous les niveaux d'utilisateurs. Administrateurs TDS, programmeurs, opérateurs maîtres et utilisateurs finals. Ce chapitre inclut des explications sur le mode de traitement des données et ressources et sur la protection, la reconstitution et l'organisation des données.
Chapitre 3	Explique la création puis l'exécution d'une application TDS. Présente la préparation et la génération de l'application. Ce chapitre inclut la description du rôle de chaque type d'utilisateur.
Chapitre 4	Décrit la conception et l'optimisation de l'application TDS. Des concepts plus complexes destinés à l'administrateur et au programmeur TDS y sont abordés.

Un glossaire des termes TDS est fourni en fin de manuel.

BIBLIOGRAPHIE

Cette bibliographie est fournie à titre indicatif. Pour plus de précisions (disponibilité du manuel, numéro de révision, indice de mise à jour), veuillez vous adresser à CEDOC.

Pour une description détaillée des concepts liés à TDS, l'utilisateur pourra consulter les manuels suivants :

Génération d'un réseau DPS 7 :

<i>Réseaux : Concepts et génération</i>	47 F2 71UC
<i>Network Overview and Generation</i>	47 A2 71UC
<i>Gestion de terminaux</i>	39 F2 24DN
<i>Terminal Management</i>	39 A2 24DN

Préface

Installation, optimisation et gestion du système :

<i>GCOS 7-V7 - Guide d'installation et de mise à jour du système</i>	47 F2 18US
<i>GCOS 7-V7 - System Installation Configuration and Updating Guide</i>	47 A2 18US
<i>GCOS 7-V7 - Manuel de l'administrateur du système</i>	47 F2 10US
<i>GCOS 7-V7 - System Administrator's Manual</i>	47 A2 10US
<i>SBR - Guide de l'utilisateur</i>	47 F2 03US
<i>System Behavior Reporter User's Guide</i>	47 A2 03US
<i>TILS - Guide de l'utilisateur</i>	47 F2 04US
<i>TILS User's Guide</i>	47 A2 04US

Création et gestion de grilles au moyen de MAINTAIN_FORM :

<i>IOF - Manuel du programmeur</i>	47 F2 37UJ
<i>IOF Programmer's Manual</i>	47 A2 37UJ

Génération du réseau GTWriter :

<i>GTWriter - Guide de l'utilisateur</i>	47 F2 05UU
<i>Generalized Terminal Writer User's Guide</i>	47 A2 05UU

Syntaxe et emploi de COBOL :

<i>COBOL 85 - Manuel de référence</i>	47 F2 05UL
<i>COBOL 85 Reference Manual</i>	47 A2 05UL
<i>COBOL 85 - Guide de l'utilisateur</i>	47 F2 06UL
<i>COBOL 85 User's Guide</i>	47 A2 06UL

Méthodes d'accès aux fichiers et gestion des données :

<i>IDS/II - Manuel de l'administrateur</i>	47 F2 13UD
<i>IDS/II Administrator's Guide</i>	47 A2 13UD
<i>UFAS-EXTENDED - Guide de l'utilisateur</i>	47 F2 04UF
<i>UFAS-EXTENDED User's Guide</i>	47 A2 04UF
<i>Fonctions de protection des données - Guide de l'utilisateur</i>	47 F2 09UF
<i>Data Security Facilities User's Guide</i>	47 A2 09UF

Tâches opérateur principal (MAIN) :

<i>GCOS7-V6 Exploitation du réseau</i>	47 F2 72UC
<i>GCOS7-V6 Network Operations Reference Manual</i>	47 A2 72UC
<i>DOF 7-PO - Guide de l'utilisateur</i>	47 F2 80UC
<i>DOF 7-PO User's Guide</i>	47 A2 80UC
<i>Guide de l'opérateur système</i>	47 F2 60UU
<i>System Operator's Guide</i>	47 A2 60UU

Gestion des accès concurrents :

<i>GAC-EXTENDED - Guide de l'utilisateur</i>	47 F2 12UF
<i>GAC-EXTENDED - User's Guide</i>	47 A2 12UF

Commandes GCL :

IOF - Manuel de référence de l'utilisateur de terminal

<i>Vol. 1 Introduction à IOF</i>	47 F2 31UJ
<i>Vol. 2 Commandes GCL</i>	47 F2 32UJ
<i>Vol. 3 Commandes processeur</i>	47 F2 33UJ
<i>Vol. 4 Commandes processeur</i>	47 F2 34UJ

IOF Terminal User's Reference Manual

<i>Part 1 Introduction to IOF</i>	47 A2 31UJ
<i>Part 2 GCL Commands</i>	47 A2 32UJ
<i>Part 3 Processors' Commands</i>	47 A2 33UJ
<i>Part 4 Appendices</i>	47 A2 34UJ

Utilisation d'IQS sous TDS :

<i>IQS/TDS-V4 - Guide de l'utilisateur</i>	47 F2 81UR
<i>IQS-V4/TDS User's Guide</i>	47 A2 81UR

Pour l'utilisation d'ORACLE sous TDS :

<i>ORACLE-V7/TDS - Guide de l'utilisateur</i>	47 F2 14UR
<i>ORACLE-V7/TDS User's Guide</i>	47 A2 14UR

Affinity :

<i>Guide d'installation et de configuration</i>	40 F2 07WA
<i>Installation & Configuration Guide</i>	40 A2 07WA

OTM :

<i>Guide de l'administrateur</i>	86 F2 24SK
<i>Administrator's Guide</i>	86 A2 24SK

CTP :

<i>CPI-C - Guide du programmeur</i>	86 F2 45SL
<i>CPI-C Programmer's Guide</i>	86 A2 45SL

Préface

IMAGEWorks :

<i>TDS - IMAGEWorks Link - Guide de l'utilisateur.....</i>	<i>47 F2 25UT</i>
<i>TDS - IMAGEWorks Link User's Guide.....</i>	<i>47 A2 25UT</i>

Dans ce manuel, les références sont citées sous forme abrégée (4 derniers caractères alphanumériques).

Pour plus de précisions, veuillez consulter le "Catalogue de la documentation" et le fascicule "Documents nouveaux".

TDS - Concepts

Table des matières

1.	Généralités	1-1
1.1	DEFINITION D'UNE APPLICATION TDS	1-2
1.1.1	Comparaison de TDS et d'autres modes de traitement	1-3
1.1.2	Avantages de TDS	1-5
1.2	TDS AU SEIN DE GCOS 7	1-6
1.2.1	Utilisateurs de TDS	1-9
1.2.2	Documentation TDS	1-10
2.	Concepts et termes fondamentaux	2-1
2.1	DEFINITION D'UNE TRANSACTION	2-2
2.1.1	Types de transactions	2-2
2.1.2	Composition d'une transaction	2-3
2.1.2.1	Définition d'un échange.....	2-4
2.1.2.2	Définition d'une TPR	2-5
2.1.2.3	Définition d'une conversation	2-5
2.2	MANIPULATION DES DONNEES ET DES RESSOURCES	2-7
2.2.1	Unité de consolidation	2-7
2.2.2	Point de consolidation	2-7
2.2.3	Types de consolidations	2-8
2.3	PROTECTION DES DONNEES ET RECONSTITUTION A PARTIR D'INCIDENTS	2-10
2.3.1	Journalisation et protection des données	2-10
2.3.1.1	Journal Avant	2-12
2.3.1.2	Journal Après	2-12

TDS - Concepts

2.3.1.3	Journal utilisateur	2-12
2.3.1.4	Utilitaires de journalisation	2-13
2.3.2	Techniques de reconstitution	2-13
2.3.2.1	Mises à jour différées	2-13
2.3.2.2	Restauration non actualisée d'une unité de consolidation	2-14
2.3.3	Restauration actualisée d'une unité de consolidation	2-15
2.4	FICHIERS	2-16
2.4.1	Fichiers contrôlés par TDS	2-16
2.4.2	Fichiers non contrôlés	2-17
2.4.3	Fichiers en ligne et hors ligne	2-18
3.	Création d'une application TDS	3-1
3.1	ETAPES DE CONSTITUTION D'UNE APPLICATION	3-2
3.2	CONCEPTION ET ELABORATION D'UNE APPLICATION TDS	3-3
3.2.1	Responsabilités de l'administrateur TDS	3-3
3.2.1.1	Préparation de fichiers en ligne et hors ligne	3-3
3.2.1.2	Ecriture du programme source TDS	3-3
3.2.1.3	Génération de l'application TDS	3-4
3.2.1.4	Catalogage des codes d'autorisation TDS	3-4
3.2.1.5	Génération de réseau	3-4
3.2.1.6	Optimisation d'une application TDS	3-6
3.2.2	Responsabilités du programmeur TDS	3-6
3.2.2.1	Ecriture des TPR	3-6
3.2.2.2	Emplacement des TPR dans une bibliothèque de modules partageables (SM)	3-8
3.2.2.3	Tests et mise au point des TPR	3-11
3.2.2.4	Définition des transactions spéciales	3-11
3.2.2.5	Création des zones de stockage d'applications	3-12
3.2.2.6	Affichage d'informations au terminal	3-13
3.2.2.7	Gestion des états	3-16
3.2.3	Assemblage des éléments de l'application	3-16
3.2.4	Modifications à posteriori d'une application TDS	3-18
3.3	EXECUTION D'UNE APPLICATION TDS	3-19
3.3.1	Responsabilités de l'opérateur maître	3-19
3.3.1.1	Lancement d'une application TDS	3-19
3.3.1.2	Utilisation de l'exploitation programmée pour gérer une application TDS	3-22
3.3.1.3	Ouverture et fermeture de fichiers	3-23
3.3.1.4	Relance après incident	3-23
3.3.1.5	Arrêt d'une application TDS	3-23
3.3.2	Responsabilités de l'utilisateur final	3-24
3.3.2.1	Lancement d'une session utilisateur final	3-24

Table des matières

3.3.2.2	Lancement des transactions	3-24
3.3.2.3	Arrêt d'une session utilisateur final.....	3-25
4.	Conception et optimisation d'une application TDS.....	4-1
4.1	CONTRÔLE DU TAUX ET DU VOLUME DE DONNÉES TRAITÉES	4-2
4.1.1	Transactions simultanées	4-2
4.1.2	Transactions non concurrentes	4-4
4.1.3	Affectation et libération de ressources.....	4-6
4.2	COMMUNICATION ENTRE CORRESPONDANTS, APPLICATIONS ET AUTRES PRODUITS GCOS 7	4-8
4.2.1	Correspondants	4-9
4.2.2	Génération dynamique de tâches à partir d'une transaction (SPAWN).....	4-10
4.2.3	Protocoles de communication entre correspondants.....	4-12
4.2.3.1	Protocole TM.....	4-12
4.2.3.2	TCAM et fonction de déport (PT) de TDS.....	4-12
4.2.3.3	Protocole XCP1.....	4-13
4.2.3.4	Le protocole XCP2.....	4-15
4.3	DIALOGUE AVEC LE MONDE UNIX.....	4-17
4.4	ACCES ET ORGANISATION DES DONNEES DANS TDS.....	4-19
4.4.1	IDS/II et TDS.....	4-19
4.4.2	Utilisation d'IQS avec TDS	4-19
4.4.3	ORACLE et TDS.....	4-20
4.5	INTEGRITÉ & SECURITE : AUTRES PRODUITS UTILISES AVEC TDS	4-21
4.5.1	TDS-HA.....	4-21
4.5.2	Disques miroirs	4-22
4.5.3	Protection des bases de données avec RDDF7	4-22
4.5.4	SECUR'ACCESS.....	4-23

Glossaire	g-1
Index	i-1

Illustrations

Figures

1-1	Environnement TDS sous GCOS 7.....	1-3
1-2	Moniteur TDS dans l'environnement TDS.....	1-4
1-3	TDS, ORACLE et IDS/II	1-6
1-4	TDS et FEPS.....	1-7
1-5	TDS et CNP 7.....	1-8
2-1	Composants d'une transaction.....	2-3
2-2	Deux types d'échanges	2-4
2-3	Conversation	2-6
2-4	Comparaison entre consolidations explicites et implicites	2-9
2-5	Journalisation TDS	2-11
2-6	Exemple de restauration non actualisée	2-14
2-7	Exemple d'une restauration actualisée	2-15
3-1	Étapes de constitution d'une application TDS	3-2
3-2	Terminaux TDS en réseau	3-5
3-3	Rapport entre les TPR et les transactions	3-7
3-4	Compilation des TPR	3-9
3-5	Segments de code et de données d'une TPR	3-10
3-6	Édition des liens des TPR	3-10
3-7	Transactions spéciales dans un TDS.....	3-12
3-8	Exemple de grille.....	3-14
3-9	Utilisation de TDS avec IMAGEWorks.....	3-15
3-10	Assemblage de TDS	3-17
3-11	Différents types de sessions	3-20
3-12	Terminal utilisateur final servant de terminal maître	3-21
3-13	Console système ou terminal IOF servant de terminal maître	3-22
3-14	Déroulement de la soumission d'une transaction	3-26
4-1	Niveau de simultanéité de 1	4-3
4-2	Transactions définies comme non-concurrentes	4-5
4-3	Unités de consolidation d'une transaction non-concurrente.....	4-6
4-4	Libération de process automatique ou non automatique	4-7
4-5	Génération dynamique	4-11
4-6	Connexion à des applications à distance avec la fonction de déport.....	4-13
4-7	Session de communication via XCP1	4-14
4-8	Session de communication via XCP2	4-16
4-9	TDS en environnement UNIX.....	4-18

TDS - Concepts

1. Généralités

Ce chapitre explique ce qu'est TDS et comprend :

- des exemples d'applications TDS,
- une description de ce qui différencie TDS des autres modes de traitement,
- une explication des avantages de TDS,
- un aperçu de la manière dont TDS dialogue avec GCOS et d'autres produits,
- une récapitulation des utilisateurs types d'une application TDS,
- une bibliographie.

Le sous-système transactionnel, ou TDS, est un environnement GCOS 7 fondé sur des transactions ou des traitements en temps réel. En effet, une application TDS accepte une demande d'un utilisateur final, la traite, puis envoie une réponse au terminal de l'utilisateur. Ce type de système est appelé transactionnel parce que le traitement est déterminé par la transaction spécifiée par un utilisateur. (Le terme "**transaction**" tel qu'il est utilisé ici, indique une interaction entre l'utilisateur final et les données stockées dans le système informatique.)

Les applications TDS permettent à plusieurs utilisateurs d'accéder aux fichiers et bases de données simultanément en préservant l'intégrité des données. Plusieurs utilisateurs peuvent même soumettre la même transaction au même moment. Les applications TDS offrent des temps de réponses courts, réduisent la durée d'indisponibilité des applications après un incident, tout en assurant la protection et la confidentialité des données.

Les noms de commandes figurant dans ce manuel sont ceux de GCOS 7 V6.

1.1 DEFINITION D'UNE APPLICATION TDS

Une application TDS est conçue pour exécuter des transactions. Le logiciel TDS agit comme gestionnaire d'applications TDS et sert à générer les applications TDS du client. Ces dernières sont créées par l'administrateur et le programmeur TDS au moyen du logiciel TDS fourni par Bull.

Dans les applications TDS, plusieurs utilisateurs ont besoin d'accéder simultanément aux informations ou de traiter des activités commerciales en continu. Pour ce faire, il existe de nombreux types d'applications TDS personnalisées. Les applications transactionnelles conviennent pour le traitement de commandes, le contrôle d'inventaire, la fabrication, la planification, la facturation, les opérations bancaires de détail et la comptabilité pour lesquelles de nombreux utilisateurs finals exploitent simultanément les mêmes programmes à partir de différents terminaux.

Exemples d'applications TDS :

Par exemple, les agents de voyage sont reliés à un ordinateur via des terminaux. Chaque agent peut consulter les informations concernant des vols et effectuer des réservations sur ceux qui l'intéressent.

Un autre exemple de transaction est le dialogue du client d'une banque avec un guichet automatique. Lorsque le client retire des espèces, plusieurs actions interviennent au cours du traitement de la transaction demandée (retrait d'espèces sur un compte) : interrogation du compte du client pour en vérifier l'approvisionnement, mise à jour du compte dans la base de données de la banque, versement du montant demandé et impression d'un reçu.

1.1.1 Comparaison de TDS et d'autres modes de traitement

TDS, comme IOF et le traitement par lots, est un environnement de GCOS 7. De même qu'avec IOF ou le traitement par lots, différentes applications TDS peuvent exister dans l'environnement TDS. La figure 1-1 décrit leurs rapports.

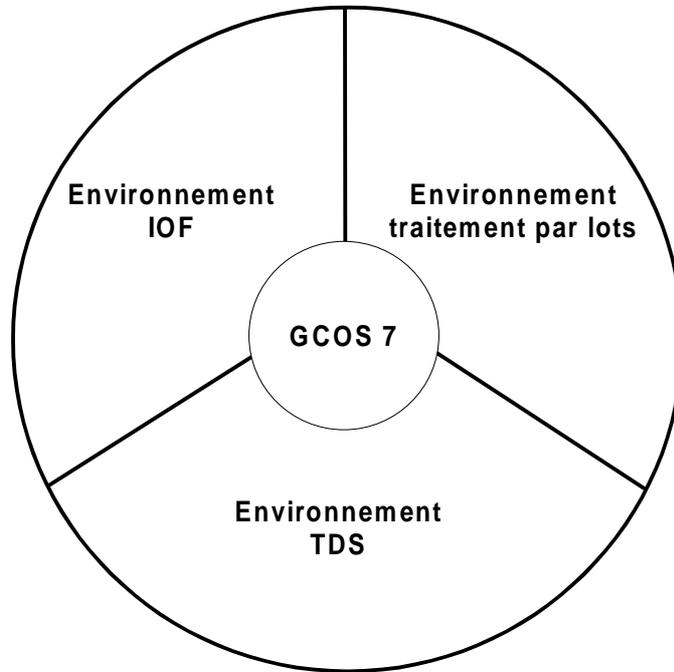


Figure 1-1. Environnement TDS sous GCOS 7

TDS satisfait rapidement les besoins d'informations des clients. Le temps nécessaire à l'application TDS pour répondre à la demande de service effectuée par un "utilisateur final" est la mesure de son efficacité. Cet intervalle est appelé "**temps de réponse**". Avec une application TDS, un utilisateur peut espérer une réponse immédiate à sa demande (1 ou 2 secondes).

En traitement par lots ou IOF, les données nécessaires à la mise à jour d'un fichier maître sont collectées avant modification. Le temps requis pour la préparation des données avant le traitement final peut être long. De plus, un fichier maître n'est utilisable que par un seul programme à la fois. Les autres programmes doivent attendre que le fichier soit libéré.

Avec une application TDS, les utilisateurs peuvent entrer des données pour un traitement immédiat et accéder aux ressources actualisées à tout moment. Le système de réservation aérienne est un exemple de traitement en temps réel. Lorsqu'un passager achète un billet, la réservation peut être immédiatement effectuée, et le système met automatiquement à jour les informations concernant la disponibilité des places. De plus, les applications TDS sont conçues pour répondre à des demandes de traitement d'informations multiples.

TDS - Concepts

TDS est une application de type interrogation/réponse. "Interrogation" signifie que les informations peuvent seulement être extraites des fichiers, sans modification du contenu de ces fichiers. "Réponse" signifie qu'une application TDS envoie les informations correspondant à une demande.

Le scénario suivant illustre l'utilité du traitement TDS :

Un utilisateur veut vérifier l'état des comptes client de sa société, et cela en mode de traitement par lots. Pour ce faire, il est nécessaire de préparer un train de travaux, planifier les travaux, imprimer et distribuer les résultats, tout un processus qui peut durer plusieurs heures. Sous TDS, l'utilisateur se connecte au système, lance une transaction et voit un menu de fonctions s'afficher à l'écran. Il sélectionne une fonction, introduit les numéros de compte et obtient l'information en quelques secondes.

Dans une application TDS, la transaction est exécutée dès son lancement. En traitement par lots ou sous IOF, les travaux sont planifiés sur un intervalle de temps, puis exécutés.

Au sein de GCOS 7, les ressources (l'environnement TDS) contenant les informations à partager par les applications TDS sont disponibles. Le moniteur TDS est le logiciel qui fournit un contrôle centralisé et ordonne le traitement des transactions (demandées par les utilisateurs à partir des terminaux). La figure 1-2 décrit la relation entre le moniteur TDS et les applications TDS.

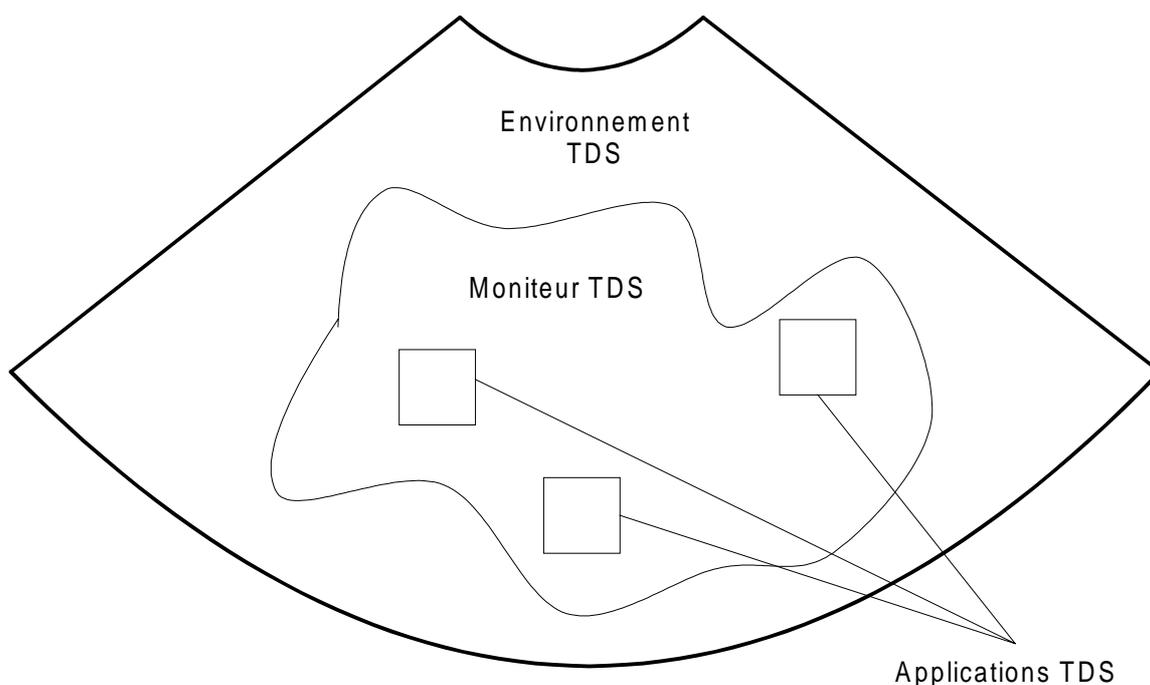


Figure 1-2. Moniteur TDS dans l'environnement TDS

1.1.2 Avantages de TDS

TDS fournit protection, intégrité et souplesse d'utilisation aux applications au moyen de ses propres mécanismes alliés à ceux de GCOS 7.

Dans une application TDS, la confidentialité des données est assurée par les codes d'autorisation TDS, les droits d'accès GCOS 7 ; elle peut être garantie par SECUR'ACCESS (produit Bull de sécurité logicielle). Le droit d'exécuter une transaction donnée peut être accepté ou refusé aux utilisateurs.

TDS permet l'exécution de plusieurs traitements en simultané (jusqu'à 124 process par application). L'intégrité et la cohérence du fichier sont assurées par les fonctions de non-concurrence, par GAC-EXTENDED et par les mécanismes de protection en lecture. Le mécanisme de reconstitution de TDS et les journaux GCOS 7 permettent de sauvegarder des données et d'effectuer des relances après arrêt prématuré.

TDS offre une souplesse de programmation. Les routines de traitement de transaction, (unités de base des transactions) peuvent être écrites en COBOL, langage C ou GPL, et peuvent intégrer les procédures IQS (système d'information relationnelle de Bull).

Le mécanisme de reconstitution de TDS, associé aux journaux du système GCOS 7, garantissent la **disponibilité** des applications. En effet, TDS permet la reprise avec relance "à chaud" en cas d'incident, la relance des transactions interrompues et la reprise dynamique sur erreur. Par ailleurs, l'utilisation de TDS avec les produits haute disponibilité (HA) de GCOS 7 améliore la protection des applications et réduit le temps d'indisponibilité de l'application :

- **TDS-HA** améliore la disponibilité des applications (système local de secours),
- la fonction **Disques miroirs** (MIRROR) améliore la disponibilité des données,
- **RDDF7** améliore la disponibilité des bases de données (système à distance de secours).

1.2 TDS AU SEIN DE GCOS 7

TDS et les bases de données

La figure 1-3 présente la relation entre TDS et les produits ORACLE et IDS/II.

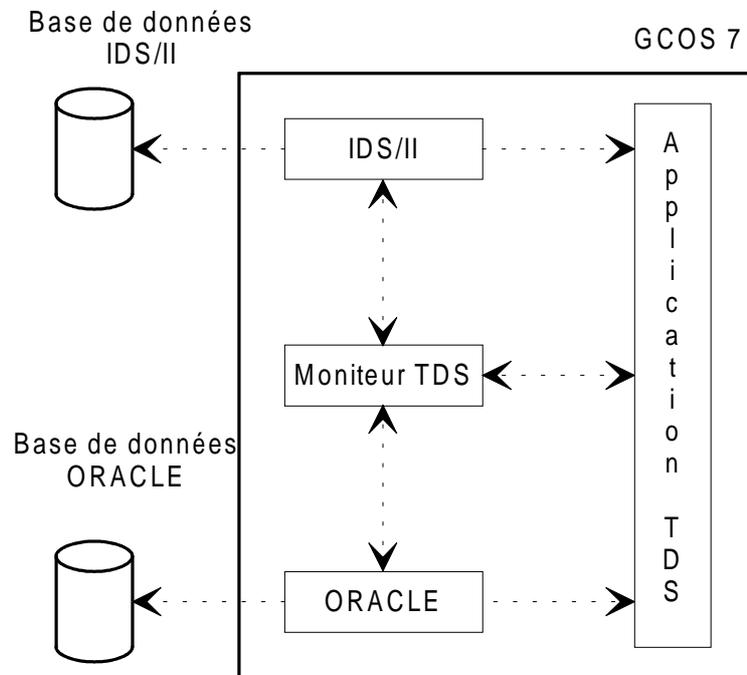


Figure 1-3. TDS, ORACLE, et IDS/II

TDS et FEPS

La figure 1-4 présente la relation entre TDS et le serveur de communications FEPS (et le système de communication directe VCAM).

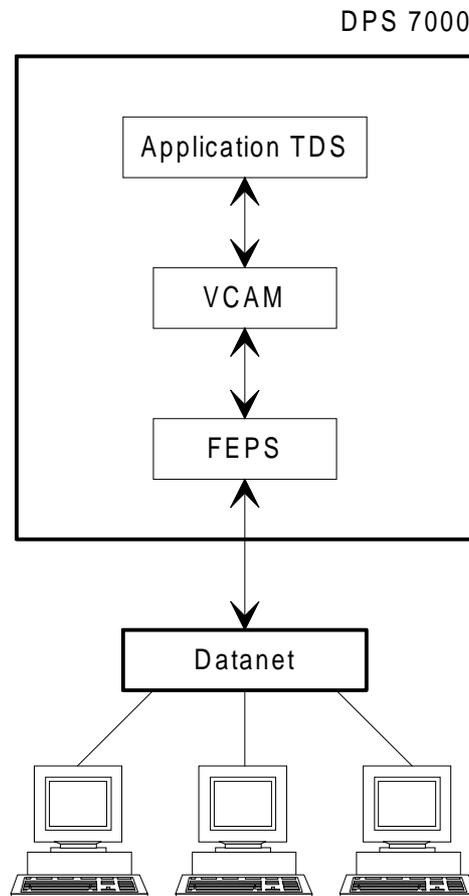


Figure 1-4. TDS et FEPS

TDS et CNP 7

La figure 1-5 présente la relation entre TDS et le frontal intégré CNP 7 (et VCAM).

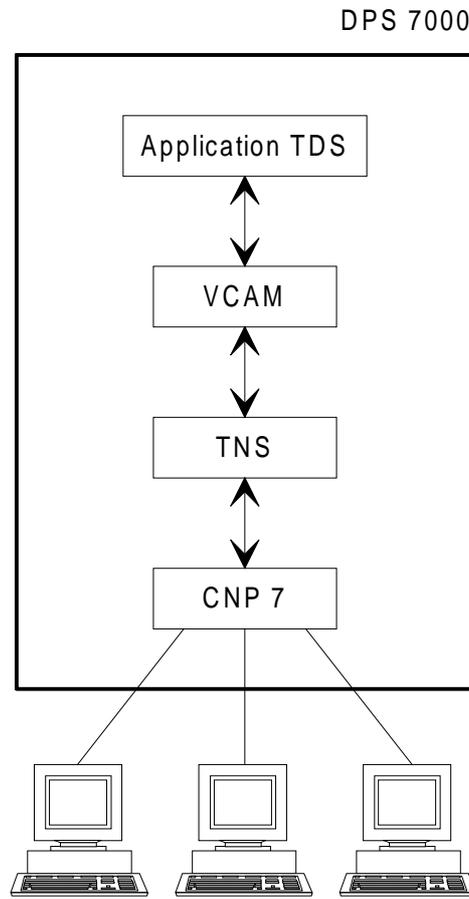


Figure 1-5. TDS et CNP 7

1.2.1 Utilisateurs de TDS

Les applications TDS sont exploitées par quatre types d'utilisateurs : l'administrateur TDS, le programmeur TDS, l'opérateur maître et l'utilisateur final.

- L'administrateur TDS est chargé de déterminer les besoins de l'application et de la personnaliser en conséquence. L'administrateur du système prépare et génère l'application, puis gère la taille et l'optimisation d'une application active.
- Le programmeur TDS écrit les transactions (ou les routines TPR) de l'application selon le schéma mis au point par l'administrateur du système.
- L'opérateur maître est chargé de la supervision d'une session TDS. L'opérateur lance et arrête l'application, ouvre et ferme les fichiers, et relance l'application en cas d'incident.
- L'utilisateur final est la personne qui, à partir de son terminal, se connecte à une application TDS qui exécute des transactions.

1.2.2 Documentation TDS

Plusieurs types de documents décrivent les divers aspects de TDS. Le type de document à consulter dépend de la fonction occupée par l'utilisateur sous TDS. La liste suivante dresse l'inventaire des documents à consulter par les quatre types d'utilisateurs de TDS. Ces documents sont également fonction des produits installés sur le site. Ceux figurant entre crochets sont facultatifs.

Utilisateur final	<i>TDS - Concepts</i>
Opérateur maître (Master)	<i>TDS - Concepts</i> <i>TDS - Guide de l'administrateur (chapitre "Commandes opérateur maître")</i>
Programmeur TDS	<i>TDS - Concepts</i> <i>TDS - COBOL ou Langage C - Guide du programmeur</i> <i>Intercommunication transactionnelle sous protocole - Guide de l'utilisateur ou XCP1/XCP2 - Guide de l'utilisateur</i> <i>TDS - Guide pratique</i> <ul style="list-style-type: none"> • <i>[IQS-V4/TDS - Guide de l'utilisateur]</i> • <i>[ORACLE-V6/TDS - Guide de l'utilisateur]</i>
Administrateur TDS	<i>TDS - Concepts</i> <i>TDS - Guide pratique</i> <i>TDS - Guide de l'administrateur</i> <ul style="list-style-type: none"> • <i>[Haute disponibilité (HA) - Guide de l'administrateur]</i>

Le programmeur TDS peut également lire le Guide de l'administrateur TDS ou du moins quelques chapitres. De même, l'administrateur TDS doit se familiariser avec une partie du contenu des manuels destinés au programmeur TDS.

Pour obtenir une bibliographie plus détaillée et les références des documents indiqués ci-dessus, se reporter à la préface du présent manuel.

2. Concepts et termes fondamentaux

Pour comprendre TDS, plusieurs concepts doivent être définis et font donc l'objet du présent chapitre :

- Transactions et éléments qui les composent :

- RoutinesTPR
 - Echanges
 - Conversations

- Gestion des données et des ressources au moyen des :

- Unités de consolidation
 - Points de consolidation

- Techniques de journalisation et de reconstitution :

- Mises à jour différées
 - Restaurations non actualisées (rollback)
 - Restaurations actualisées (rollforward)

- Fichiers des applications TDS :

- Fichiers contrôlés par TDS
 - Fichiers non contrôlés par TDS

2.1 DEFINITION D'UNE TRANSACTION

Sous TDS, une transaction est une séquence d'activités destinées à exécuter une tâche utilisateur. Une application TDS peut comporter jusqu'à 2000 transactions. Une transaction (Tx) est désignée par un identificateur de message spécifique (id-message). Ce dernier est fourni par l'administrateur TDS ou affiché dans un menu.

L'introduction d'un id-message par l'utilisateur provoque l'exécution ou le traitement de la transaction. Cette dernière prend fin lorsque la tâche a été entièrement exécutée. Lorsque le moniteur TDS reçoit un id-message à partir d'un terminal, il charge la première TPR de la transaction. Une transaction peut comprendre une ou plusieurs TPR. Si tout le travail d'une transaction n'est pas accompli par la première TPR, d'autres TPR sont lancées.

L'utilisateur ne peut introduire qu'un seul id-message à la fois, à partir d'un terminal donné. Si l'exécution de la transaction réussit, il peut entrer l'id-message suivant.

2.1.1 Types de transactions

Sous TDS, il existe trois principaux types de transactions.

- La transaction FOR INQUIRY est une demande d'information. Elle répond à la question de l'utilisateur en "interrogeant" la base de données sur l'état courant des données, sans modifier aucun fichier.
- La transaction UPDATE est une modification de fichiers ou bases de données. Elle exécute la demande de l'utilisateur en ajoutant ou supprimant les données demandées, et modifie donc les fichiers.
- La transaction FOR DEBUG peut être considérée comme temporaire. Elle permet au programmeur de résoudre un problème ou d'optimiser une application TDS.

Par ailleurs, il existe plusieurs types de transactions spéciales réservées au programmeur TDS. Pour plus d'informations à ce sujet, se référer au paragraphe "Responsabilités du programmeur TDS" dans le chapitre "Création d'une application TDS".

2.1.2 Composition d'une transaction

Une transaction comporte :

- des échanges,
- des routines TPR et
- des conversations.

La figure 2-1 montre la relation entre ces éléments.

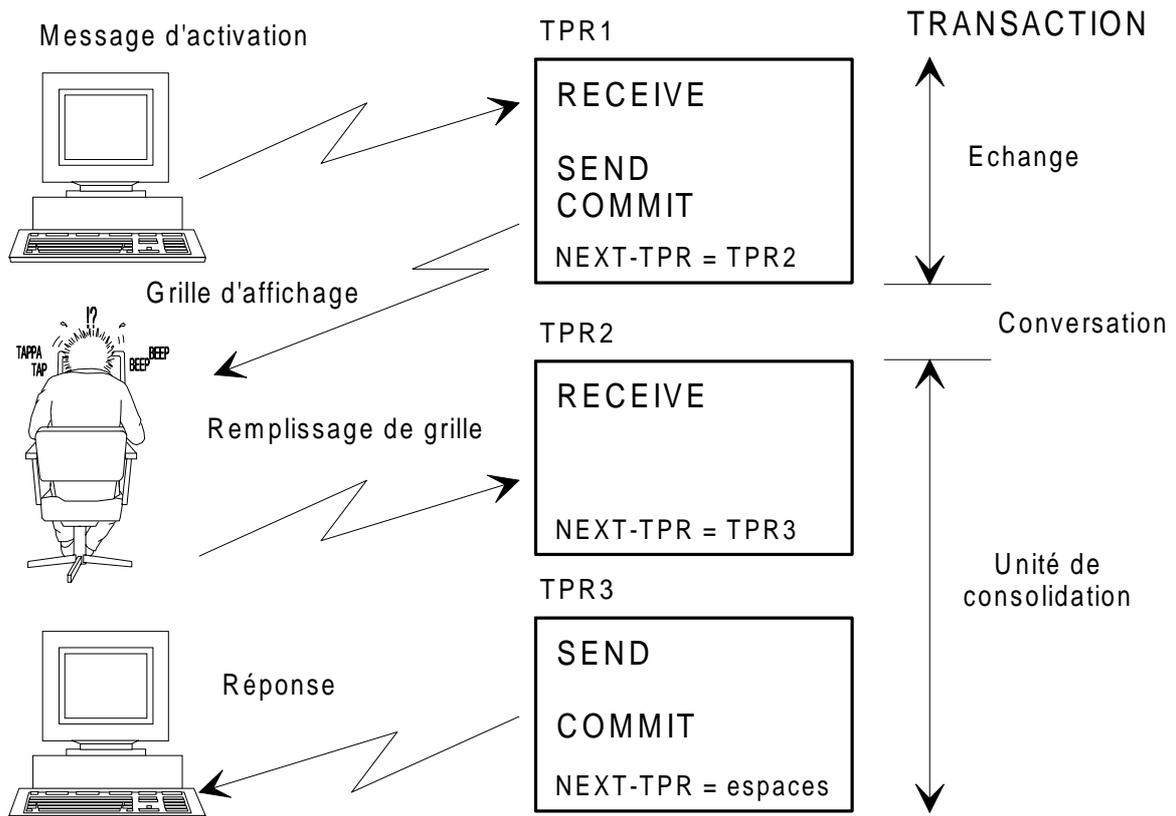


Figure 2-1. Composants d'une transaction

Une transaction est une séquence d'échanges et de conversations. Elle comprend au moins un **échange**, lequel contient une ou plusieurs TPR. Cependant, une transaction peut ou non inclure une conversation. Le traitement d'une transaction est effectué par les TPR au cours d'un échange, et la réponse de l'opérateur intervient pendant la conversation.

2.1.2.1 Définition d'un échange

Chaque échange comprend un message d'entrée, un traitement et un message de sortie. Le traitement qui intervient pendant l'échange est exécuté par une ou plusieurs **routines de traitement de transaction (TPR)**.

Un échange débute à la transmission d'un message émis par l'opérateur de terminal et se termine à la réponse qui lui est envoyée. Un échange est une séquence qui se déroule comme suit :

- demande introduite par l'utilisateur (INPUT),
- traitement exécuté par la transaction,
- réponse envoyée au terminal de l'utilisateur (OUTPUT).

Un échange contient au moins une TPR et peut en comporter plusieurs liées entre elles. La figure 2-2 présente ces deux possibilités.

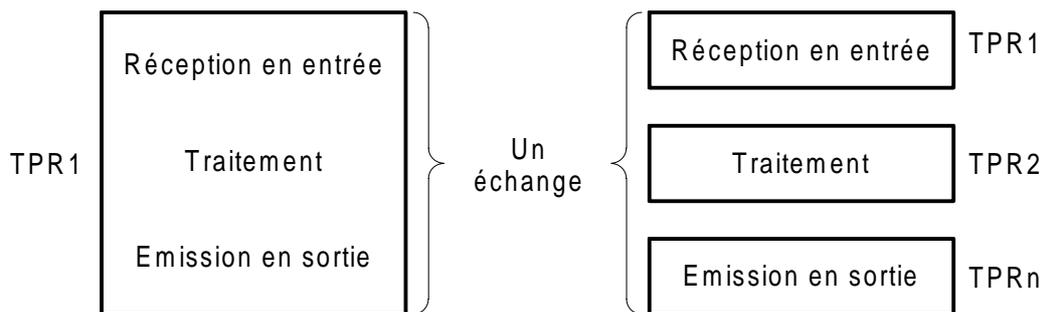


Figure 2-2. Deux types d'échanges

Lorsqu'une transaction est lancée, le moniteur TDS charge la première TPR. Le traitement de la TPR, c'est-à-dire l'entrée et la sortie, constitue un échange. Si la transaction est interactive, c'est à dire si un utilisateur final doit répondre à un message affiché à l'écran, une conversation s'instaure. Après réception de la réponse de l'opérateur de terminal, la TPR suivante est chargée puis exécutée.

A la fin d'un échange, le contrôle est rendu au terminal. La fin de l'échange est signalée soit par la réponse transmise à l'opérateur, soit par la réponse implicite, "READY", lorsque la transaction se termine sans message à l'écran. Après introduction des données, l'opérateur doit attendre la réponse de la transaction avant d'introduire un nouveau message.

2.1.2.2 Définition d'une TPR

Une TPR est une routine de traitement de transaction écrite en COBOL, en C ou en langage de programmation GPL. Les TPR sont des programmes qui exécutent les différentes étapes d'une transaction. Cette dernière comprend une ou plusieurs TPR associées qui servent à recevoir les données en entrée, à les traiter, puis à émettre la réponse.

Généralement, les TPR sont chaînées. Lorsqu'une TPR n'est pas chaînée avec une autre, il s'agit de la dernière TPR de la transaction.

Le programmeur TDS écrit et compile les TPR, puis les stocke dans des bibliothèques de modules partageables (SM). Les TPR sont mises en mémoire principale pour s'exécuter au besoin. Différentes transactions peuvent partager la même TPR, dans n'importe quel ordre (à savoir, ordre d'exécution) avec d'autres routines TPR. Toute application TDS peut comporter plusieurs TPR s'exécutant simultanément.

Un système peut disposer au maximum de 40 000 TPR ; chaque application TDS peut en comporter jusqu'à 25 200.

2.1.2.3 Définition d'une conversation

Sous TDS, une conversation est le laps de temps compris entre deux échanges au cours de la même transaction. Une conversation correspond au temps écoulé entre la réception d'un message de sortie par l'utilisateur et l'émission par ce dernier d'un message d'entrée. Cette période commence avec la réponse envoyée au terminal par une transaction, couvre le temps de réflexion de l'utilisateur (temps mort) et prend fin à la réception du message introduit ensuite par l'utilisateur.

La figure 2-3 illustre une conversation.

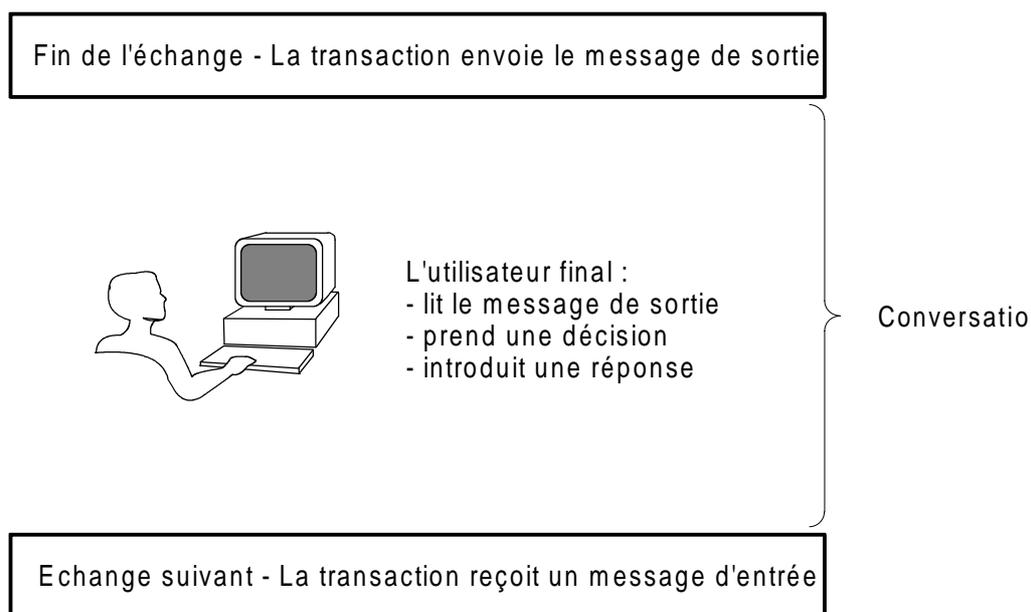


Figure 2-3. Conversation

La rapport conversation/échange est le suivant :

- Une transaction ne comporte pas nécessairement de conversation.
- Le "temps mort" entre le message d'entrée et la réponse fait partie de la conversation. Il peut être assez long, spécialement si l'utilisateur doit effectuer une intervention, ou si la transaction implique deux personnes (par exemple un vendeur et un client).

Une conversation exige plus de temps que le traitement d'un échange, car elle nécessite une intervention humaine. L'application TDS peut employer le temps d'attente à exécuter d'autres fonctions, il s'agit là d'un point dont le programmeur TDS doit tenir compte lorsqu'il crée les TPR.

2.2 MANIPULATION DES DONNEES ET DES RESSOURCES

Une transaction n'a pas accès exclusif aux données utilisées par ses TPR. Si une transaction mobilise toutes les données (ou ressources) auxquelles elle accède, les autres transactions ne peuvent pas y accéder tant qu'elle n'est pas terminée.

A un certain point du traitement, une transaction doit "sauvegarder" les modifications effectuées sur les données. Cette sauvegarde est appelée "consolidation". Le moment de la consolidation au cours de la transaction s'appelle point de consolidation.

2.2.1 Unité de consolidation

Une unité de consolidation (UC) est une unité de traitement d'une transaction. Elle comprend l'ensemble du traitement depuis le dernier point de consolidation (ou depuis le début de la transaction s'il s'agit de la première unité de consolidation) jusqu'au point de consolidation courant (ou en fin de transaction s'il s'agit de la dernière unité de consolidation). Les opérations d'écriture, de suppression, de lecture ou de mise à jour de fichiers s'effectuent en fonction de cette unité de traitement. Lorsqu'une consolidation est effectuée, les modifications apportées pendant l'unité de consolidation deviennent permanentes. La consolidation permet aux autres transactions (lancées par d'autres utilisateurs) d'accéder aux mêmes données. Les consolidations permettent de réduire le nombre des transactions en attente de ressources, ainsi que les temps de réponse. Des points de consolidation bien placés permettent d'améliorer la rapidité, la performance et la sécurité du traitement des données.

Le programmeur TDS définit le moment où doit s'effectuer la consolidation au cours de la transaction. Après chaque unité de consolidation, les modifications des fichiers ou bases de données sont enregistrées. Une fois cette opération effectuée, les ressources sont disponibles pour d'autres transactions.

2.2.2 Point de consolidation

Un point de consolidation est le point (ou position) auquel doit s'effectuer une consolidation au cours de l'exécution d'une transaction. Une transaction se déroule toujours entre deux points de consolidation ; celui qui précède et celui en cours. Au niveau du point de consolidation, les conditions suivantes sont réunies :

- tous les traitements (opérations sur les fichiers, messages, demandes de soumission de travaux) qui précèdent sont définitivement terminés et ne peuvent pas être annulés,
- du point de vue de l'utilisateur, les données et fichiers sont dans un état cohérent,
- les transactions peuvent être relancées, en cas d'incident logiciel ou matériel. Dans ce cas, toutes les données enregistrées avant le dernier point de consolidation sont fiables.

A noter que, le début d'une transaction est, de façon implicite, un point de consolidation.

Au point de consolidation, TDS libère les ressources affectées à la transaction. (Pendant le traitement d'une transaction, les données sont transférées des disques à la mémoire principale et sont conservées dans des objets appelés tampons et pages.)

Les points de consolidations permettent de résoudre les conflits entre des TPR qui demandent les mêmes ressources. Si une transaction ne peut pas poursuivre le traitement en raison d'un tel conflit, elle est arrêtée prématurément, puis relancée au point de consolidation précédent. Dans l'intervalle, la TPR qui détient la ressource l'aura libérée à son point de consolidation.

2.2.3 Types de consolidations

Il existe deux types de consolidations : la consolidation explicite (sous contrôle du programmeur TDS) et la consolidation implicite (sous contrôle de TDS). A la génération de l'application TDS, l'administrateur TDS définit, pour chaque transaction, le type de consolidation qui sera utilisé.

Consolidation implicite

Si l'administrateur TDS demande la consolidation implicite pour une transaction, celle-ci a lieu automatiquement :

- à chaque conversation,
- à la fin de la transaction,
- à la fin d'une TPR si le programmeur TDS a spécifié un délai maximum en secondes entre la fin de la TPR et le début de la suivante (valeur de WAIT-TIME).

Certaines commandes permettent au programmeur d'annuler toutes les consolidations implicites, à l'exception de celles qui ont lieu en fin de transaction.

Consolidation explicite

Les consolidations explicites sont sous le contrôle du programmeur. Pour chaque consolidation en cours de transaction, le programmeur doit ajouter une instruction dans le programme TPR. La consolidation s'effectue à la fin de la TPR courante.

La figure 2-4, présentée sous forme de colonnes, compare les consolidations implicites et explicites et explique comment le programmeur peut passer d'un mode de consolidation automatique à un autre (implicite ou explicite).

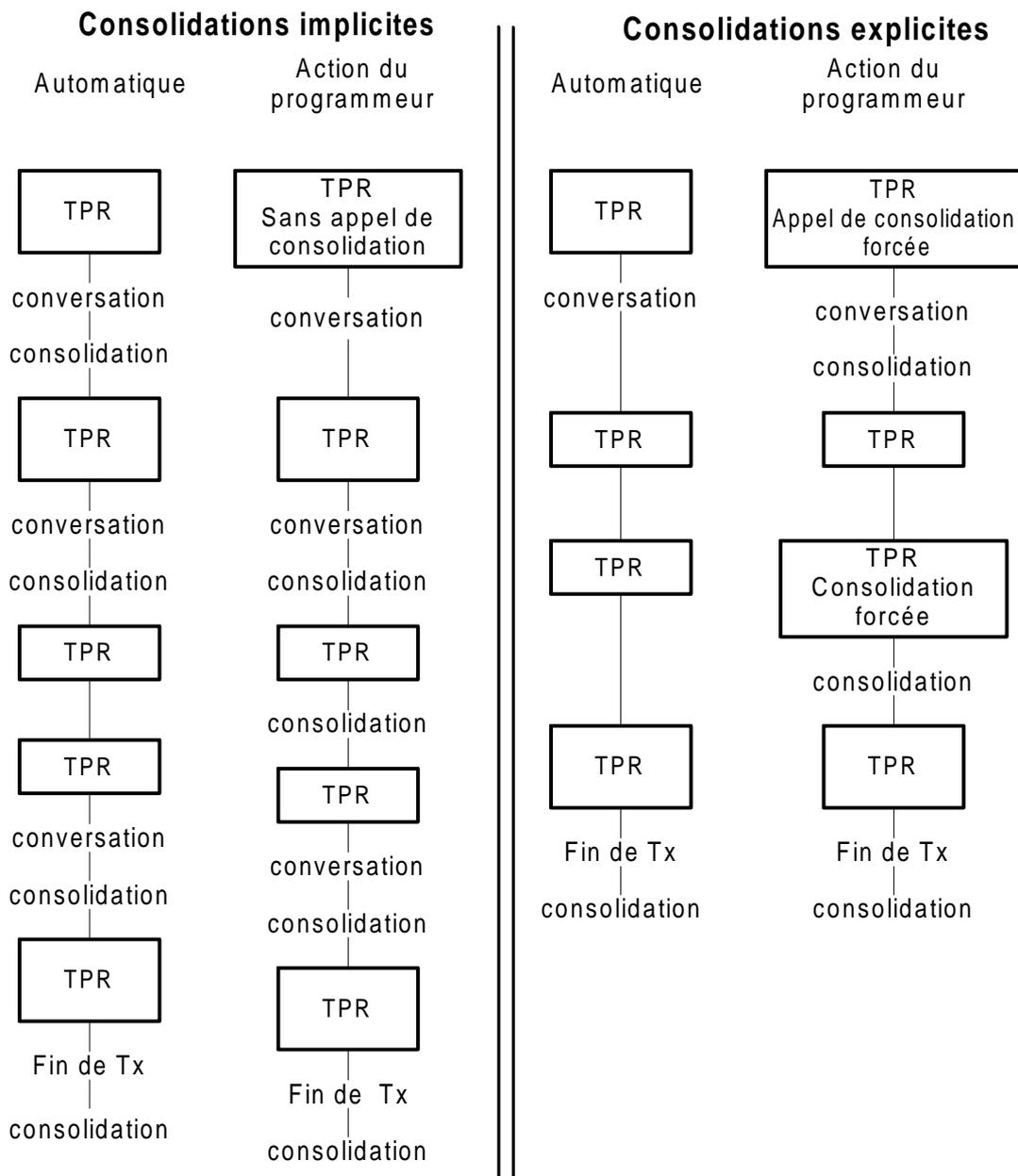


Figure 2-4. Comparaison entre consolidations explicites et implicites

Les modifications apportées par une TPR aux fichiers ou bases de données ne sont effectives qu'après consolidation.

2.3 PROTECTION DES DONNEES ET RECONSTITUTION A PARTIR D'INCIDENTS

TDS fournit différents mécanismes de protection des données en cas d'incidents matériels ou logiciels. Les mécanismes de journalisation et de reconstitution garantissent l'intégrité des données en cas d'arrêt prématuré d'une transaction ou de TDS, ou en cas de crash système de GCOS 7. Les procédures d'intégrité et de reconstitution varient suivant le type d'incident et le niveau de protection demandé.

Par exemple, si un crash système se produit pendant l'exécution d'une transaction qui transfère l'argent d'un compte à un autre, les données "mises à jour" sont incohérentes et probablement incomplètes. Lors d'un incident, la fiabilité des données mises à jour n'est assurée que s'il y a eu consolidation. La journalisation TDS et les mécanismes de reconstitution servent à sauvegarder les unités de consolidation et donc à gérer ce type de situation.

2.3.1 Journalisation et protection des données

La journalisation consiste à utiliser des journaux pour préserver l'intégrité des données. Deux journaux GCOS 7 assurent cette fonction ; un autre fichier journal TDS peut être personnalisé et utilisé à des fins statistiques et de mise au point. Ces trois journaux sont :

- le journal Avant,
- le journal Après,
- le journal utilisateur.

Les journaux Avant et Après ne sont pas propres à TDS, ce sont des fonctions de GCOS 7. Le journal utilisateur est une fonction TDS.

La journalisation consiste à effectuer deux copies des données en cours de traitement. Ces copies sont appelées "images". La première image des données est effectuée avant la mise à jour des données. Elle est appelée **image Avant**. La seconde image des données, ou **image Après**, est effectuée après la mise à jour des données. Ces images sont respectivement enregistrées, ou journalisées dans le journal Avant, dans le journal Après, et dans les journaux TDS.

Le journal Avant permet la reconstitution de fichiers lorsque les mises à jour sont incomplètes suite à un incident logiciel. Le journal Après protège les fichiers contre tout type de destruction, y compris la destruction physique du support de fichier.

L'administrateur TDS définit le niveau de protection pour chaque application. Le journal utilisé varie suivant que le fichier est protégé contre un incident matériel, logiciel ou les deux. La protection d'un fichier par journalisation peut être spécifiée soit au cours de son traitement, soit dans la description du catalogue, si le fichier est catalogué.

La figure 2-5 montre l'intervention des journaux Avant et Après dans une transaction comprenant deux échanges et une mise à jour de fichier.

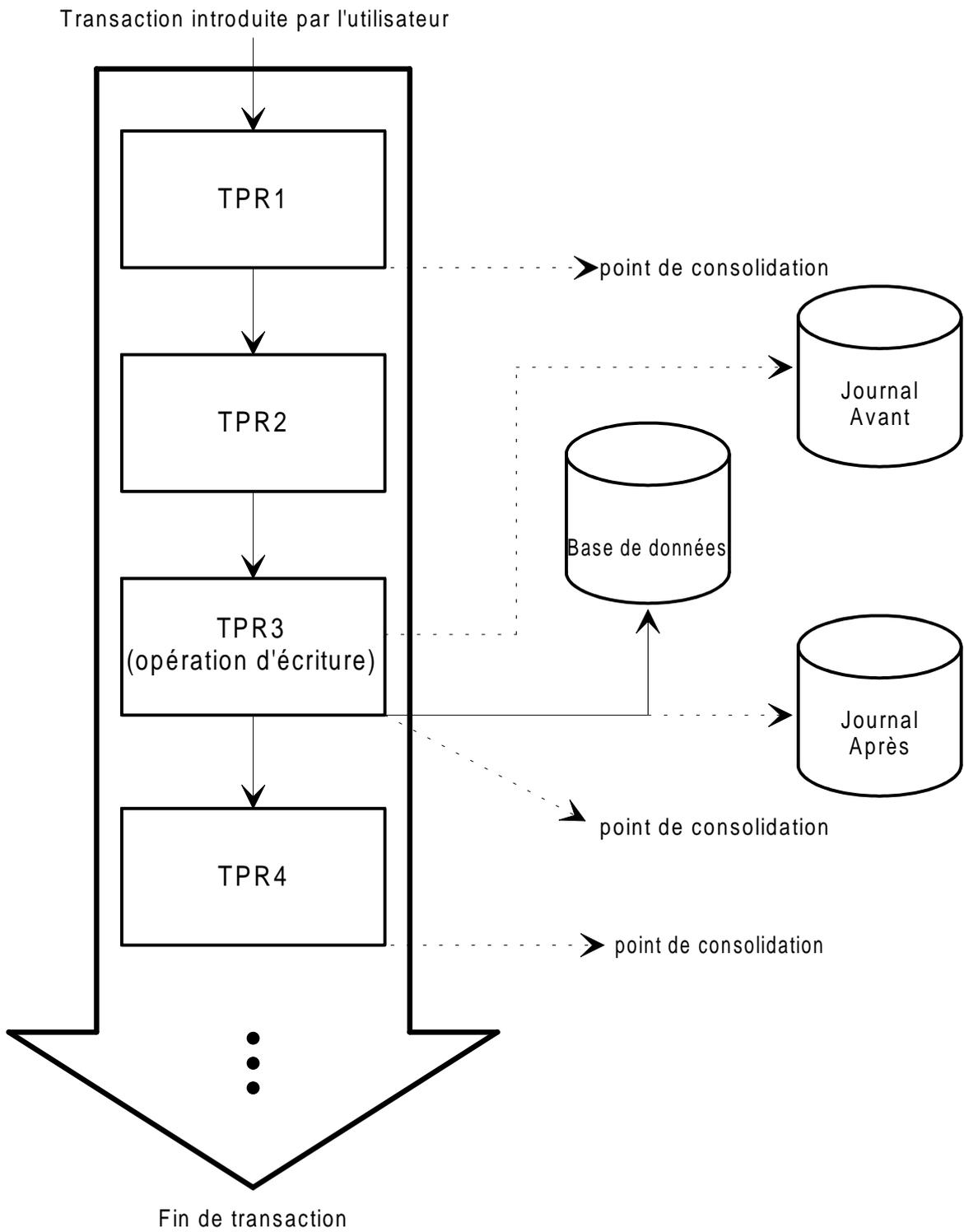


Figure 2-5. Journalisation TDS

2.3.1.1 Journal Avant

Le journal Avant protège les fichiers contre les incidents logiciels. Il ne les protège pas contre la destruction physique suite à un incident matériel. Le journal Avant sert à reconstituer les fichiers lorsque des mises à jour sont arrêtées prématurément en cours de traitement.

Le journal Avant conserve une image du fichier à mettre à jour. Les images sont stockées dans le journal Avant et permettent de ramener les fichiers à leur état correct le plus récent, c'est-à-dire au début de la dernière unité de consolidation. Les images du journal Avant sont prises avant le début de la mise à jour, pendant que s'effectue l'unité de consolidation. Une fois celle-ci réalisée, les images Avant sont libérées.

Le journal Avant doit être utilisé pour les fichiers partagés entre une application TDS et des programmes de traitement par lots. Il peut être utilisé seul ou associé au journal Après.

2.3.1.2 Journal Après

Le journal Après protège les données contre les incidents matériels et logiciels. Il sert à garder trace des modifications apportées au fichier au cas où un incident se produirait entre la mise à jour et son enregistrement dans le fichier.

Lorsqu'une unité de consolidation actualise un fichier, le système conserve une image de chaque article après sa mise à jour, et enregistre cette "image Après" dans le journal Après. En cas d'incident, il est possible de reconstituer le fichier au moyen de la version sauvegardée précédemment, puis de remplacer chaque article mis à jour par son image Après.

La sauvegarde et la reconstitution de fichiers ne sont pas automatiques sous TDS. L'opérateur maître ou l'administrateur doivent exécuter ces tâches au moyen des utilitaires de gestion de données.

2.3.1.3 Journal utilisateur

Sous TDS, un troisième journal permet aux utilisateurs d'enregistrer des statistiques concernant une application TDS. Un utilisateur peut décider d'y consigner des messages d'entrée et de sortie, ainsi que des informations sur les transactions.

Le journal utilisateur se trouve dans le journal Après. Il ne s'agit pas d'un mécanisme de protection de fichier, mais il est mentionné ici en raison de son emplacement. Cependant, en cas de besoins, les informations qu'il contient peuvent être transférées dans un fichier particulier et utilisées à des fins de mise au point.

2.3.1.4 Utilitaires de journalisation

Il existe d'autres utilitaires associés au journal Après. Ils sont réservés à l'administrateur TDS :

- L'utilitaire ROLLFWD utilise le journal Après pour ramener des fichiers (1 à 25) à l'état dans lequel ils se trouvaient à une date et une heure spécifiées.
- L'utilitaire DUMPJRNL prend les données du journal utilisateur et les enregistre dans un fichier de sortie séquentiel.
- L'utilitaire de reconstitution du journal Après (JRU) ramène le journal Après à un état cohérent s'il est altéré car TDS ne peut pas fournir la liste des unités de consolidations arrêtées prématurément.

2.3.2 Techniques de reconstitution

Les techniques de reconstitution TDS ramènent les données à l'état qui était le leur au moment de l'incident. Les mécanismes de reconstitution doivent exécuter ces tâches rapidement et facilement afin que le traitement des transactions en cours puissent reprendre.

Trois mécanismes de reconstitution sont utilisés dans TDS :

- mises à jour différées,
- restaurations non actualisées (rollback),
- restaurations actualisées (rollforward).

Ces mécanismes sont utilisés en association avec les journaux.

2.3.2.1 Mises à jour différées

Le mécanisme de mise à jour différée fonctionne avec le journal Après. Si une mise à jour différée est requise pour un fichier, l'instruction WRITE ou REWRITE de la TPR n'est exécutée physiquement qu'après la fin normale de l'unité de consolidation. Pendant le traitement de l'unité de consolidation, l'article de sortie est mis en tampon dans le journal Après. Si l'unité de consolidation n'aboutit pas, les modifications sont ignorées.

L'enregistrement physique dans le fichier s'effectue selon les modalités suivantes :

- Si l'unité de consolidation s'arrête prématurément, le système efface le contenu du tampon, de même que les images Après. Aucune modification n'est appliquée au fichier.
- Si l'unité de consolidation est validée, le contenu du tampon est transféré dans le fichier.

Avec la mise à jour différée, les modifications apportées aux fichiers sont acceptées, mais ne sont réellement enregistrées sur disque que lorsque le point de consolidation est atteint. Si une unité de consolidation n'aboutit pas, les données TDS n'en sont pas affectées.

2.3.2.2 Restauration non actualisée d'une unité de consolidation

Lorsqu'un incident se produit au cours du traitement d'une unité de consolidation, cette dernière doit être relancée. Ceci s'effectue au moyen d'une "restauration non actualisée". De cette manière, les fichiers journalisés sont ramenés à leur état correct, au début de la dernière unité de consolidation.

Avec une restauration non actualisée, les images Avant sont copiées dans les fichiers correspondants, ce qui les ramène à leur état antérieur. Autrement dit, lorsqu'aucune consolidation n'est effectuée, les données sont ramenées à leur état initial par application des images Avant. La transaction peut être relancée, et traiter les mêmes données comme si elles étaient utilisées pour la première fois.

La figure 2-6 présente une restauration non actualisée. Dans cet exemple, la première unité de consolidation se termine normalement (UC1). En revanche, un arrêt prématuré se produit au cours de la seconde unité de consolidation (UC2). Lorsqu'une restauration non actualisée est exécutée, les fichiers retournent à l'état dans lequel ils se trouvaient à la fin de la première unité de consolidation.

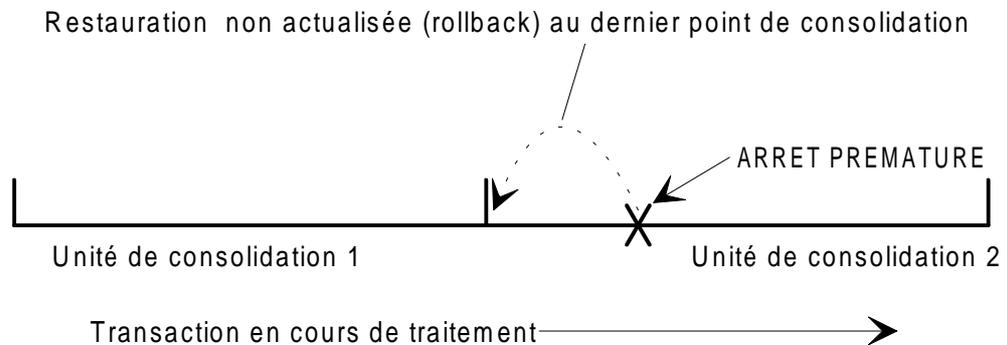


Figure 2-6. Exemple de restauration non actualisée

2.3.3 Restauration actualisée d'une unité de consolidation

Lorsqu'un incident se produit après une consolidation, mais avant enregistrement des données dans le fichier, le transfert d'E/S vers le fichier doit être relancé. (L'opérateur en est avisé lorsque la mise à jour du fichier échoue). Dans ce cas, la restauration actualisée utilise les images Après pour mettre à jour le fichier.

Si l'incident provient d'un problème matériel, celui-ci doit être traité avant que la mise à jour du fichier soit effectuée par application des images Après.

La restauration actualisée met à jour les fichiers journalisés en les ramenant à leur état valide, tel qu'il est déterminé par la dernière unité de consolidation (voir figure 2-7). Dans cet exemple, la première unité de consolidation se termine normalement, mais le transfert des modifications dans le fichier s'arrête prématurément. Puisque les mises à jour ont été consolidées, une restauration actualisée peut s'appliquer au fichier.

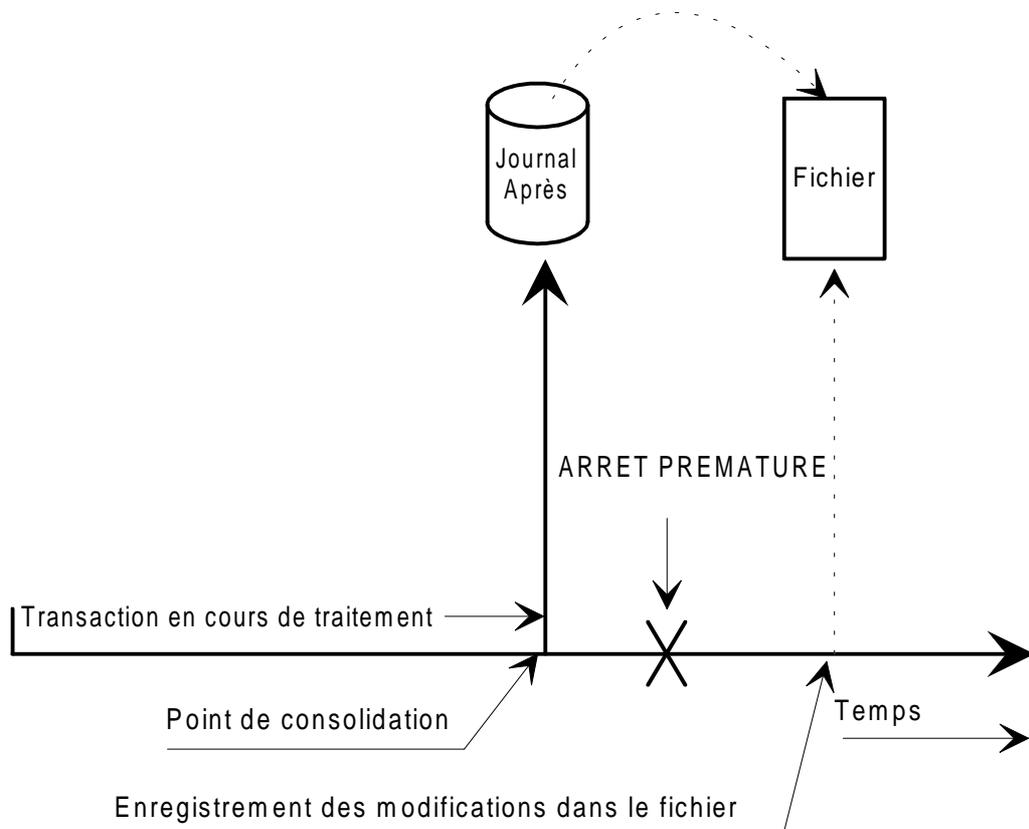


Figure 2-7. Exemple d'une restauration actualisée

2.4 FICHIERS

Dans une application TDS, il existe deux types de fichiers. Les fichiers contrôlés par TDS et les fichiers non contrôlés par TDS. Ceux-ci diffèrent par leur méthode de résolution des conflits d'accès. Dans les deux cas, il s'agit de fichiers UFAS-EXTENDED. Chaque type de fichiers est cependant soumis à certaines restrictions.

En cas de conflit d'accès lié à un fichier contrôlé par TDS, le conflit est résolu par le moniteur TDS. Lorsqu'un fichier non contrôlé par TDS est utilisé, il est "verrouillé" (mais uniquement pendant le temps "d'accès") pour empêcher d'autres transactions d'y accéder.

2.4.1 Fichiers contrôlés par TDS

L'administrateur TDS peut déclarer qu'un fichier UFAS-EXTENDED sera contrôlé par TDS. Il peut s'agir d'un fichier séquentiel indexé ou relatif. Lorsqu'un fichier est contrôlé par TDS, l'accès simultané aux articles du fichier est contrôlé par le moniteur TDS.

Lorsqu'une transaction accède aux articles d'un fichier, une partie du fichier est verrouillée pour la durée de l'unité de consolidation. L'accès aux articles verrouillés est interdit aux autres transactions. Dans ce cas de figure, la notion de fichiers contrôlés par TDS prend toute sa mesure, tous les conflits d'accès étant résolus par le moniteur TDS.

Par exemple, deux agences de réservation aérienne peuvent essayer simultanément d'attribuer à deux passagers la même place sur un vol. Si le fichier des réservations est contrôlé par TDS, l'accès concurrent aux fichiers est résolu. L'accès à des données identiques demandé simultanément par les deux utilisateurs est possible, mais de façon contrôlée.

Le mécanisme de contrôle des accès concurrents, connu sous le nom de GAC-EXTENDED, permet à un fichier contrôlé par TDS d'être partagé entre une transaction et :

- une autre transaction de la même application TDS,
- une autre application TDS,
- un programme en traitement par lots,
- une application IQS.

Les aires IDS/II doivent être déclarées sous contrôle de TDS. Sous GAC-EXTENDED, les fichiers contrôlés par TDS et les aires IDS/II peuvent être affectés exclusivement ou non au travail TDS. Un accès exclusif signifie que l'accès est fourni uniquement à la TPR qui le demande.

L'emploi de GAC-EXTENDED est spécifié au moment de l'affectation des fichiers. GAC-EXTENDED s'applique aux fichiers catalogués ou non. Dans ce dernier cas, il est possible de définir l'emploi de GAC-EXTENDED dans les ordres JCL d'exécution du travail TDS.

Par ailleurs, les fichiers contrôlés par TDS peuvent être protégés au moyen d'utilitaires de journalisation étendus.

2.4.2 Fichiers non contrôlés

Les fichiers non contrôlés par TDS sont également de type UFAS-EXTENDED. Un fichier non contrôlé peut être protégé par le journal Avant, mais non par le journal Après. Ce type de fichier n'est pas protégé par GAC-EXTENDED. Par conséquent, des conflits d'accès peuvent se produire, par exemple entre deux TPR de la même application TDS.

Contrairement aux fichiers contrôlés par TDS, la protection des fichiers non contrôlés nécessite l'intervention d'un programmeur TDS. Ce dernier doit veiller à ce qu'un fichier non contrôlé ne fasse pas l'objet d'un accès simultané par une autre transaction (de la même application ou d'une autre).

Différentes méthodes permettent de protéger les fichiers non contrôlés. Le programmeur peut :

- affecter le fichier exclusivement à une application TDS,
- faire en sorte que les transactions qui ont accès à un même fichier ne s'exécutent pas en même temps,
- protéger les fichiers non contrôlés par un mécanisme de verrouillage indirect fondé sur une identification utilisateur (composée de lettres ou de chiffres). Les utilisateurs peuvent partager des fichiers non contrôlés en associant un fichier à un nom donné, puis en verrouillant ce nom. Cette technique doit être programmée dans les TPR. Les fonctions de verrouillage/déverrouillage des TPR permettent aux utilisateurs d'empêcher temporairement l'accès au fichier par d'autres transactions, que ce soit en écriture ou en lecture. Lorsque l'utilisateur en a terminé avec le fichier, le nom correspondant au fichier verrouillé est déverrouillé. Il s'agit là d'une méthode de verrouillage indirecte (les fichiers réels sont verrouillés par l'utilisation de chiffres ou lettres), rien ne garantit donc son efficacité constante.

L'opérateur maître dispose d'une autre méthode de verrouillage de fichiers. Il peut, en effet, bloquer le lancement d'une transaction ou d'une classe de transactions.

2.4.3 Fichiers en ligne et hors ligne

Les fichiers en ligne et hors ligne sont destinés à générer et traiter l'application TDS. Ils peuvent être considérés comme des "fichiers système" TDS. Il ne s'agit pas de fichiers utilisateur formant la base de données d'une application. L'utilisation de ces fichiers est présentée au chapitre "Création d'une application TDS". Pour plus d'informations concernant les fichiers individuels, voir le *Guide de l'administrateur - TDS*.

Fichiers hors ligne

Les fichiers hors ligne permettent de générer, compiler et relier TDS. Ils sont nécessaires à l'exécution. Les fichiers TDS hors ligne sont les suivants :

<nom-tds>.SLLIB	Bibliothèque origine contenant le programme origine de génération de TDS et les commandes de l'éditeur de liens.
<nom-tds>.COBOL	Bibliothèque origine contenant les descriptions de fichier et zones de stockage définies à la génération.
<nom-tds>.SMLIB	Bibliothèque de modules partageables contenant les modules édités des TPR.
<nom-tds>.LMLIB	Bibliothèque contenant le module chargeable de génération de TDS.
<nom-tds>.EDITION	Fichier de travail contenant le compte rendu produit par le programme de génération de TDS.

Fichiers en ligne

Il s'agit de fichiers servant au traitement des transactions. Ils doivent donc être en ligne, ou accessibles, pendant que l'application est active. Les fichiers en ligne TDS sont les suivants :

<nom-tds>.SWAP	Fichier journal interne TDS contenant les informations de relance pour chaque terminal connecté à TDS.
<nom-tds>.CTLM	Fichier (contrôlé par GAC-EXTENDED) contenant les informations de gestion lors de la génération ; celles-ci sont mises à jour au cours de l'exécution de l'application.
<nom-tds>.CTLN	Fichier (non contrôlé par GAC-EXTENDED) contenant les informations de gestion lors de la génération ; celles-ci sont mises à jour au cours de l'exécution de l'application.
<nom-tds>.RECOV	Fichier contenant les informations nécessaires à une restauration non actualisée ou actualisée dynamique des fichiers contrôlés par TDS.
<nom-tds>.DEBUG	Fichier contenant les résultats des options de suivi (TRACE).
<nom-tds>.PPCLOG	Fichier utilisé par le service XCP2 et contenant l'état courant des pools.
<nom-tds>.GMEM	Fichier contenant une description des objets IQS.
<nom-tds>.H_MMS	Fichier contenant les informations qui permettent l'utilisation d'IMAGEWorks.

3. Création d'une application TDS

La création d'une application TDS opérationnelle implique différentes procédures. Ces responsabilités sont prises en charge par l'administrateur TDS, le programmeur TDS, l'opérateur maître et l'utilisateur final.

Ce chapitre décrit les tâches à exécuter et détermine les responsabilités de chaque type d'utilisateur pendant la préparation.

3.1 ETAPES DE CONSTITUTION D'UNE APPLICATION

L'environnement TDS et le programme de génération sont fournis en standard. L'administrateur et le programmeur TDS doivent personnaliser les applications en fonction des spécifications du site. La figure 3-1 montre les principales opérations à effectuer pour personnaliser une application TDS. L'utilisateur responsable est indiqué en regard de chaque activité.

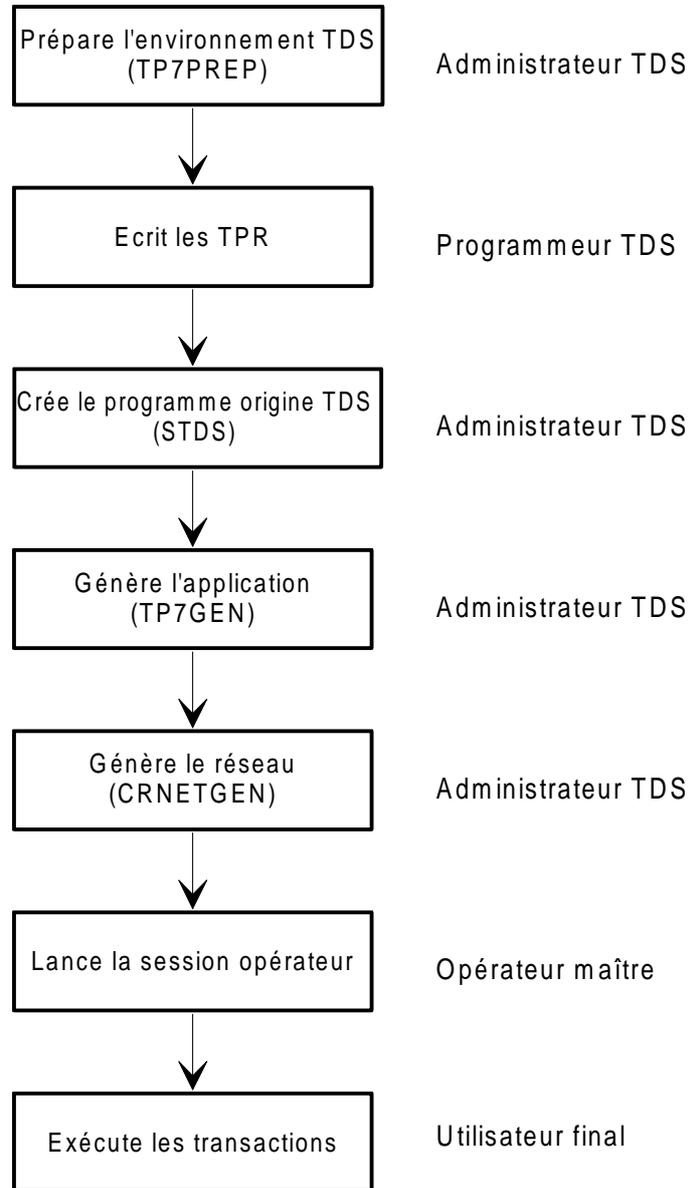


Figure 3-1. Etapes de constitution d'une application TDS

3.2 CONCEPTION ET ELABORATION D'UNE APPLICATION TDS

L'administrateur et le programmeur TDS sont chargés de la conception et de la création d'une application TDS. Une fois ces tâches effectuées, l'opérateur maître peut lancer le TDS et l'utilisateur final peut commencer à exécuter des transactions.

3.2.1 Responsabilités de l'administrateur TDS

L'administrateur TDS est chargé de planifier et de définir l'application, de la configurer, puis de surveiller la gestion d'une application en exécution. L'administrateur crée une application selon les spécifications du site en déclarant les fichiers, les transactions à utiliser, les structures de stockage et les correspondants pouvant accéder à l'application.

3.2.1.1 Préparation de fichiers en ligne et hors ligne

Les fichiers en ligne et hors ligne sont les "fichiers système" d'une application TDS. Ils sont destinés à la création de l'environnement TDS et au traitement des transactions. Ces fichiers ne sont pas utilisés pour manipuler des données utilisateur d'une application.

L'administrateur TDS emploie l'utilitaire JCL TP7PREP pour créer les fichiers en ligne et hors ligne. TP7PREP est un outil logiciel qui aide l'administrateur à construire l'application en fournissant des fichiers déjà préparés. Avec cet utilitaire, l'administrateur définit les fichiers, leur réserve de la place, spécifie leur taille ainsi que le support et le type d'appareil. Si les tailles de fichier implicites ne conviennent pas, l'administrateur peut spécifier d'autres tailles.

3.2.1.2 Ecriture du programme source TDS

Le programme source TDS (STDS) est destiné à la génération de l'application. C'est dans ce programme que l'administrateur définit l'application. Une fois que ce programme est écrit, l'administrateur le sauvegarde dans le fichier hors ligne <nom-tds>.SLLIB.

Le programme STDS se compose de trois sections :

- TDS SECTION
- INPUT-OUTPUT SECTION
- TRANSACTION SECTION

Dans la section TDS, l'administrateur spécifie les contraintes générales de TDS. Dans la section INPUT-OUTPUT, il donne les informations concernant l'accès aux fichiers, y compris les descriptions de fichiers, leurs modes de traitement, leur protection et les références de schéma de base de données. Dans la section TRANSACTION, l'administrateur détaille toutes les transactions auxquelles les utilisateurs peuvent accéder, les ressources disponibles pour les transactions et les contraintes à respecter.

L'administrateur doit ordonner les instructions du programme STDS comme elles apparaissent dans le *Guide de l'administrateur TDS*. Pour plus d'informations sur les instructions utilisées dans le programme STDS, voir le *Guide de l'Administrateur TDS*.

3.2.1.3 Génération de l'application TDS

Une fois que l'administrateur a réservé de la place aux fichiers requis et stocké le programme origine (STDS) dans le fichier hors ligne <nom-tds>.SLLIB, le programme de génération de TDS (TDSGEN) doit être exécuté. TDSGEN est exécuté au moyen de l'utilitaire JCL TP7GEN.

Cet utilitaire formate et initialise les "fichiers système" TDS, compile le programme STDS, et prépare les fichiers de commande utilisés lors de l'édition de liens de TPR. Il crée également les unités (dans <nom-tds>.COBOL) qui sont copiées dans les TPR.

3.2.1.4 Catalogage des codes d'autorisation TDS

L'administrateur TDS doit déclarer les informations concernant les utilisateurs dans le catalogue de site (SITE.CATALOG). Ce dernier contient une liste des utilisateurs TDS associés aux projets. Une liste de codes d'autorisation est attribuée à chaque projet. Les codes d'autorisation déterminent si un projet peut accéder ou non à un ensemble de transactions particulier.

En établissant des codes d'autorisation, l'administrateur définit pour chaque opérateur le jeu de transactions qu'il peut exécuter, éliminant de ce fait le risque d'accès non autorisé. Par exemple, si l'administrateur TDS donne à une transaction uniquement le code d'autorisation 10, pour utiliser la transaction l'opérateur doit être affecté à un projet ayant le code d'autorisation 10.

3.2.1.5 Génération de réseau

A l'intérieur d'un TDS, les terminaux communiquent souvent avec l'application à partir de plusieurs sites. Le programme de communications de données sur le réseau doit être décrit indépendamment de TDS. L'objectif de la description de réseau est de refléter le nombre d'utilisateurs et les types de correspondants déclarés à la génération de TDS.

Si le réseau de communications n'est pas encore défini, l'administrateur doit le générer après la génération de TDS et compilation/édition de liens de ses transactions. La génération de réseau (NETGEN) est exécutée au moyen de l'utilitaire de génération de réseau CRNETGEN (Create Network Generation). Lors de la génération du réseau, l'administrateur en définit la configuration et déclare le nombre d'utilisateurs TDS ainsi que les types de correspondants.

La figure 3-2 présente une application TDS en réseau.

Création d'une application TDS

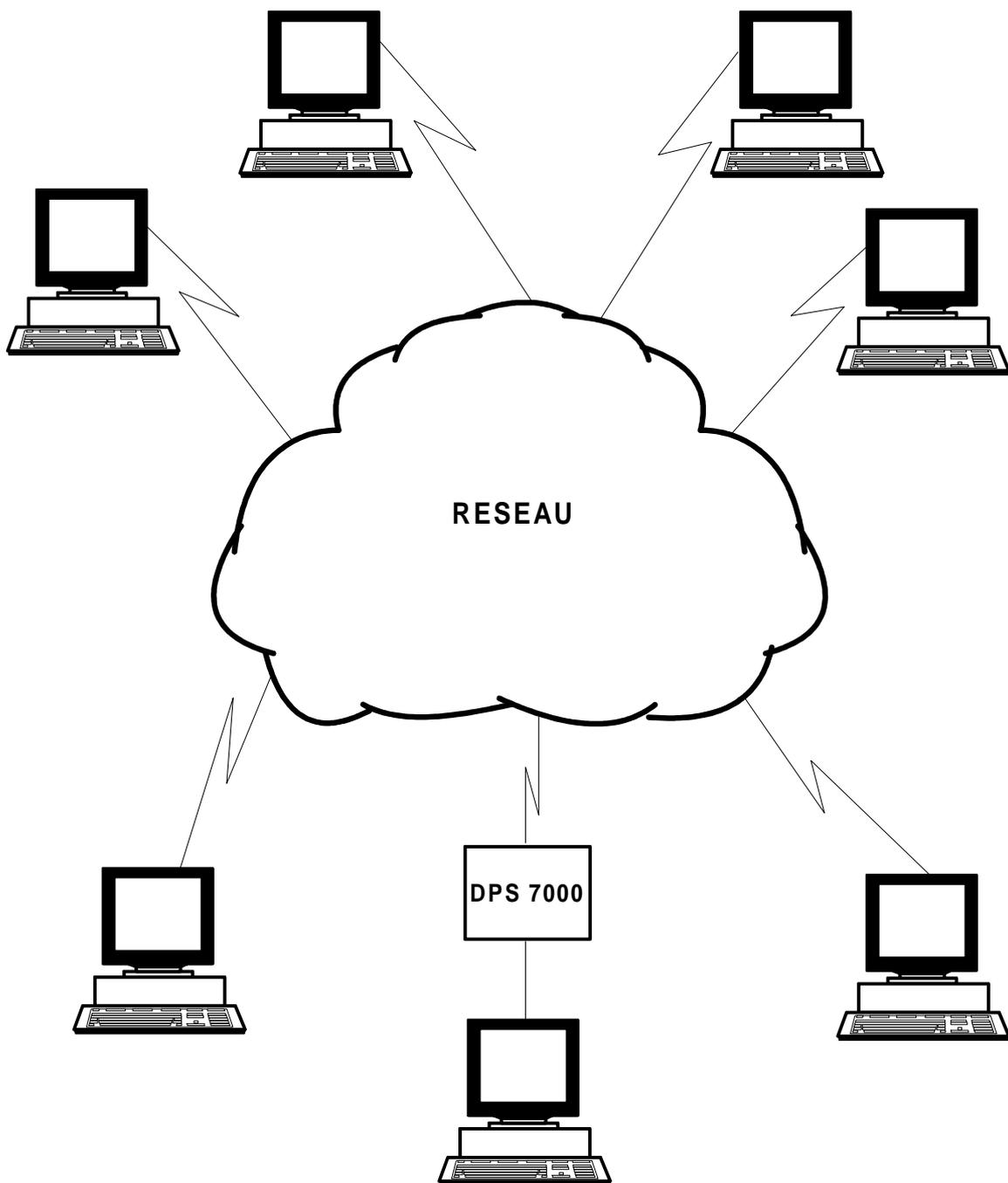


Figure 3-2. Terminals TDS en réseau

Après génération du réseau, la session de communications, puis la session TDS, peuvent être lancées.

3.2.1.6 Optimisation d'une application TDS

Pour mesurer les performances et optimiser les ressources du système, l'administrateur peut utiliser les deux fonctions suivantes :

- TILS
- SBR.

Le logiciel de surveillance du système SBR est destiné à analyser la charge de travail et l'utilisation des ressources matérielles et logicielles d'une installation DPS 7000. Il peut servir à mesurer l'efficacité des TPR, ainsi que le comportement de GAC-EXTENDED et d'un TDS. Pour plus d'informations, se référer au *Guide de l'utilisateur SBR*.

Le simulateur de charge transactionnelle et interactive TILS est destiné à simuler une charge de travail TDS. Il peut être utilisé sur le même système que l'application, ou sur un système à distance. TILS sert à tester les applications en modification ou développement. Pour plus d'informations, se référer au *Guide de l'utilisateur TILS*.

3.2.2 Responsabilités du programmeur TDS

Le programmeur TDS est responsable des opérations suivantes :

- écriture des TPR (routines de traitement de transaction),
- écriture des transactions spéciales,
- déclaration des zones de stockage TDS,
- définition du mode d'affichage des informations à l'écran,
- gestion des états,
- tests et mise au point des TPR et des applications.

3.2.2.1 Ecriture des TPR

Le programmeur TDS peut commencer à écrire le code origine des routines de traitement de transaction (TPR) avant que l'administrateur TDS génère l'application. Les TPR sont des programmes utilisateur écrits pour une application. Elles peuvent être écrites en COBOL, langage C ou GPL. Les TPR exécutent les tâches ci-après :

- lecture de fichiers,
- traitement des données,
- mise à jour des fichiers,
- dialogue avec l'utilisateur final qui a lancé la transaction.

Pour plus d'informations sur les TPR, se référer au paragraphe "Définition d'une TPR" dans le chapitre "Concepts et termes fondamentaux".

La figure 3-3 décrit le rapport des TPR avec une transaction. Les routines TPR1 à TPR4 sont fonctionnellement dépendantes de la transaction 2 (Tx2). Dans cet exemple, ces TPR ne peuvent pas être utilisées par les transactions 1 (Tx1) et 3 (Tx3). Toutefois, dans une application réelle, les TPR sont partagées par de nombreuses transactions différentes.

Création d'une application TDS

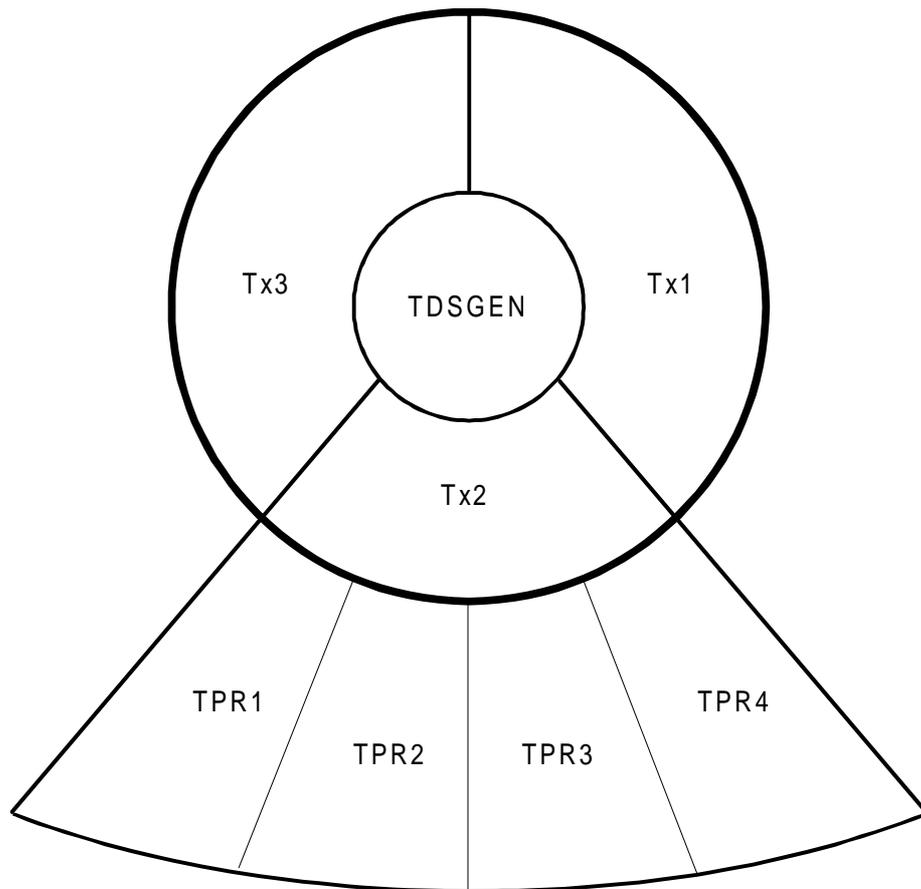


Figure 3-3. Rapport entre les TPR et les transactions

Le programmeur TDS doit déterminer les transactions nécessaires à une application, puis écrire les TPR correspondant à ces transactions. L'écriture des TPR inclut le codage, l'édition, la compilation, l'édition de liens et les tests des TPR.

Le programmeur TDS doit concevoir et écrire les TPR de façon à optimiser les temps de réponse. Les ressources ne doivent pas être mobilisées en attente d'une réponse de l'opérateur de terminal. Les TPR performantes utilisent puis libèrent les ressources. Les fonctions doivent être affectées entre différentes TPR. Si une TPR traite trop de tâches, les temps de réponse augmentent, les ressources sont bloquées et des conflits se produisent.

L'erreur la plus commune dans la conception de TDS consiste à ne créer qu'une TPR pour réaliser l'ensemble d'une transaction. Par exemple, une TPR peut être utilisée pour mettre à jour différents articles d'un ensemble restreint de fichiers. Toutefois, une TPR qui lit ou met à jour des dizaines d'articles n'est pas performante. Les TPR ainsi conçues entrent en conflit avec les autres transactions qui ont besoin d'accéder à ces articles, ou occupent un process TDS pendant longtemps, entraînant des temps d'attentes pour les autres utilisateurs.

Le moniteur TDS fournit des procédures de gestion des TPR, des télécommunications et des données.

Le programmeur TDS peut décrire les fichiers et les zones de stockage des transactions dans le programme de génération de TDS et faire en sorte que les TPR copient les descriptions à l'aide de la fonction COPY en COBOL. Cela permet de créer une structure cohérente pour une application TDS, de réduire également le temps consacré au codage des TPR et d'éviter les erreurs.

Conseils pour l'écriture des TPR

- Concevoir des transactions correspondant à une unité de base de traitement.
- Ecrire des TPR concises et compactes.
- Déterminer les opérations indivisibles.
- Constituer chaque opération indivisible en une seule unité de consolidation.
- Libérer les ressources au début d'une conversation. Une conversation requiert plus de temps que le traitement d'un échange.
- Ecrire des TPR conformément à un plan. Ecrire des TPR indépendamment les unes des autres peut conduire à des erreurs et à des incohérences.
- Créer des TPR spéciales pour le traitement d'éléments "exceptionnels" (TPR de traitement d'erreurs, par exemple).

3.2.2.2 Emplacement des TPR dans une bibliothèque de modules partageables (SM)

Le programmeur TDS compile toutes les TPR séparément, puis effectue l'édition des liens en exécutant l'utilitaire LINKER. Cet utilitaire crée des références entre les unités compilées (CU) et établit les liens avec les procédures TDS. Lorsque l'édition des liens est effectuée, les TPR sont stockées dans une bibliothèque de modules partageables (SMLIB).

Après compilation des TPR origine, des unités compilées contenant des segments de code et de données des TPR sont produites (voir figures 3-4 et 3-5).

Création d'une application TDS

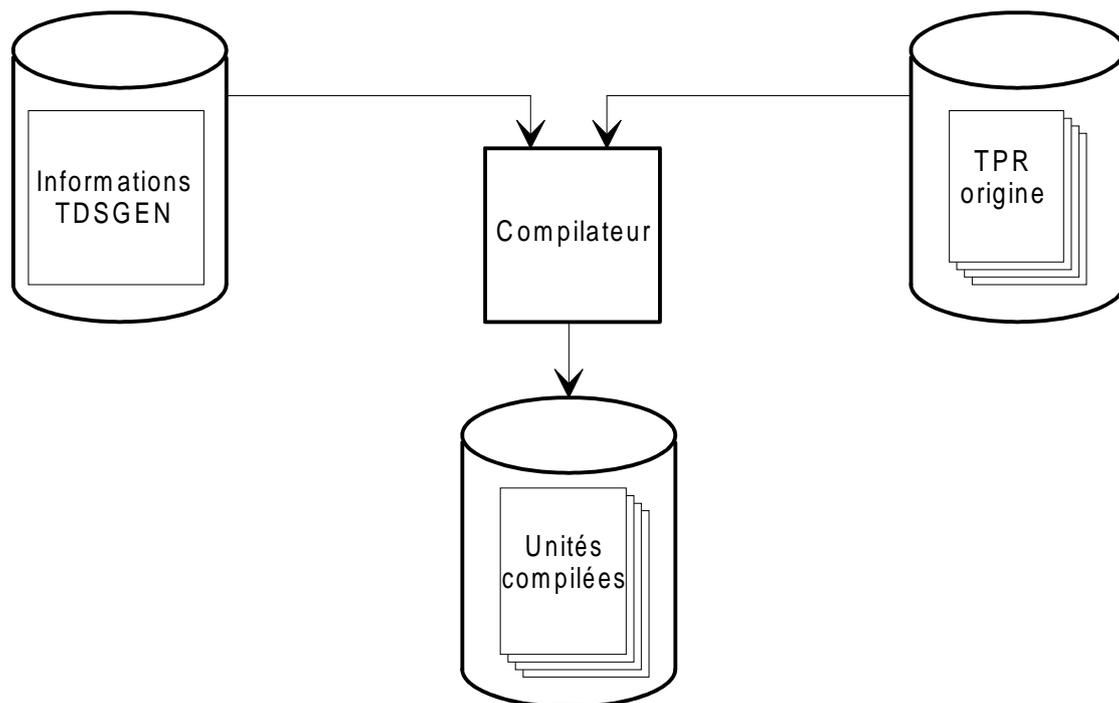


Figure 3-4. Compilation des TPR

Chaque TPR comprend des segments de code et de données comme illustré (voir figure 3-5).

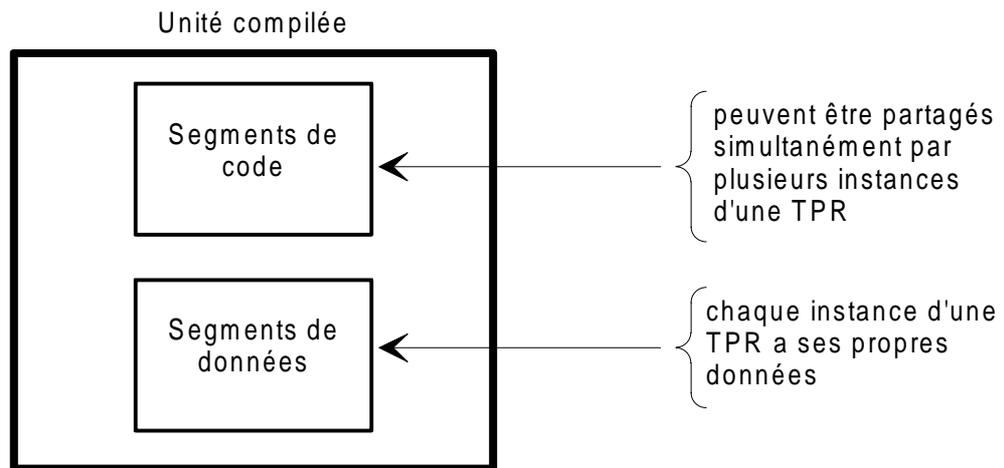


Figure 3-5. Segments de code et de données d'une TPR

A l'exécution de l'utilitaire LINKER, chaque TPR compilée est stockée dans un module partageable (figure 3-6).

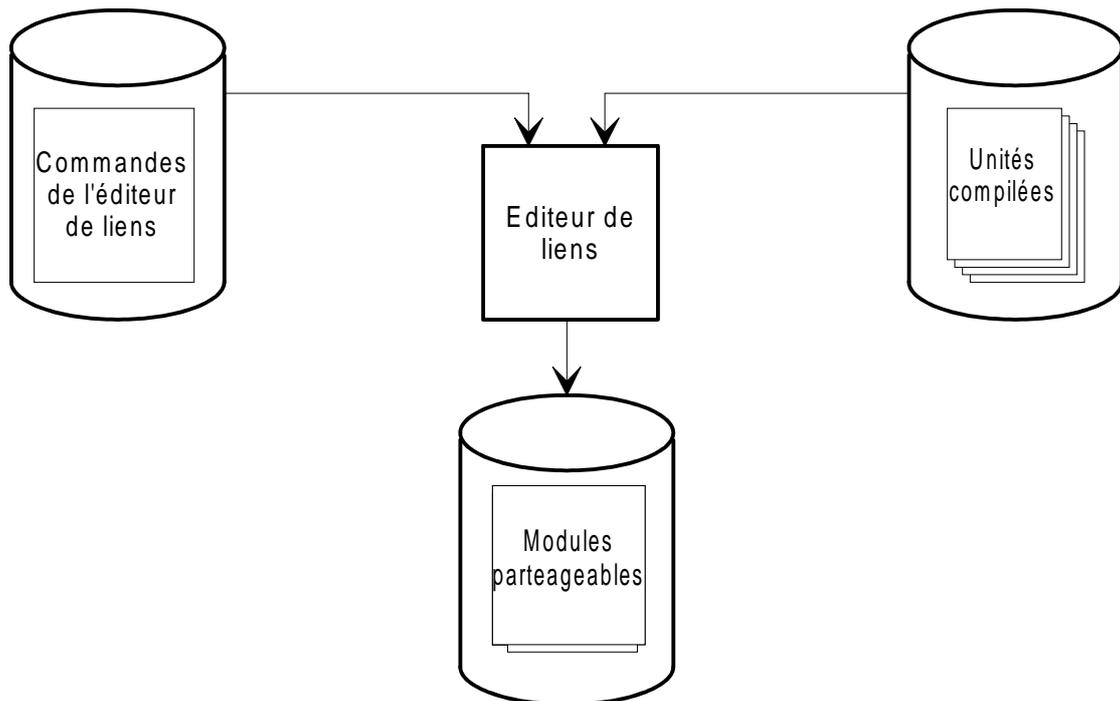


Figure 3-6. Edition des liens des TPR

Les TPR sont des entités partageables. Ceci signifie que différentes transactions peuvent simultanément utiliser la même TPR. Si une nouvelle transaction est activée et demande une TPR en cours d'exécution, TDS crée des segments de données de la TPR à l'état initial pour cette nouvelle transaction. Les segments de code n'ont pas besoin d'être copiés puisqu'ils sont partageables.

3.2.2.3 Tests et mise au point des TPR

Il existe différents moyens de tester et de mettre au point des TPR et des applications TDS :

- Utiliser la commande TRACE pour mettre au point des TPR. Cette commande accepte un sous-ensemble des commandes de PCF (programme d'aide à la mise au point).
- Spécifier le mode DEBUG dans le programme de génération TDS. Lorsque les transactions sont exécutées en mode DEBUG, les mises à jour de fichiers sont annulées. Les TPR peuvent donc être exécutées et testées sans modifier aucune donnée de façon permanente.
- Simuler une application TDS opérationnelle. L'interface traitement par lots de TDS (Batch Interface) est un programme conçu pour simuler un environnement TDS complètement opérationnel sans devoir utiliser les terminaux. Les programmes de l'interface traitement par lots de TDS peuvent être écrits par le programmeur, en COBOL, et utilisés pour simuler les fonctions de l'opérateur de terminal.

Le programme de traitement par lots peut se connecter comme un terminal pour envoyer et recevoir des messages. L'application TDS le considère comme un terminal connecté, mais sans les contraintes en temps réel d'un terminal.

- Utiliser les statistiques en temps réel pour afficher les informations détaillées et les statistiques générales sur les correspondants et fichiers d'une application TDS active. Ceci permet au programmeur d'identifier les problèmes (utilisateurs bloqués, fichiers ouverts ou fermés, etc.) et d'effectuer les corrections nécessaires.

3.2.2.4 Définition des transactions spéciales

Il existe sept transactions spéciales implicites destinées à contrôler une application TDS :

- STARTUP
- LOGON
- RESTART
- BREAK
- DISCNCT (For DISCONNECT)
- LOGOUT
- SHUTDOWN.

La figure 3-7 les situe dans le contexte d'une application.

Dans une session TDS, la mise en route et l'arrêt du système ne sont effectués qu'une seule fois.

Toutefois, durant une session utilisateur, il peut y avoir plusieurs demandes d'entrée en communication. Par exemple, si un utilisateur est déconnecté et se reconnecte ensuite.

L'utilisateur peut vouloir provoquer une interruption (par exemple, en appuyant sur une touche BREAK), ce qui interrompt la transaction en cours et provoque le lancement de la transaction BREAK.

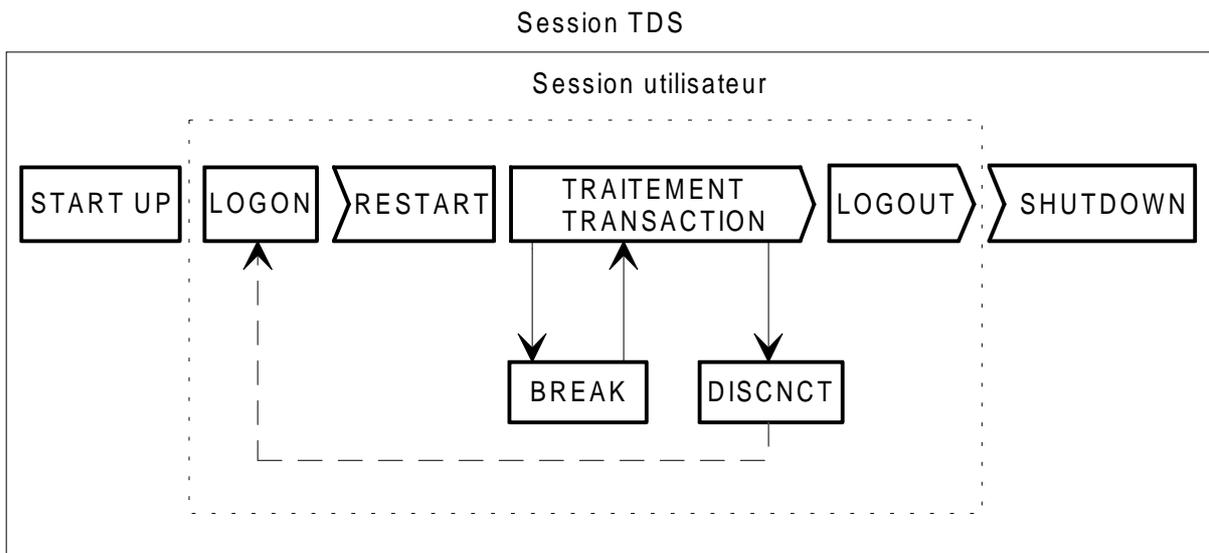


Figure 3-7. Transactions spéciales dans un TDS

Le programmeur TDS peut remplacer les transactions implicites par des transactions spéciales personnalisées. Il est souhaitable d'écrire des transactions personnalisées lorsque les transactions implicites ne conviennent pas pour un site ou une application donnés. Pour plus d'informations sur ces transactions, se référer au *Manuel du programmeur TDS*.

3.2.2.5 Création des zones de stockage d'applications

Dans une application TDS, il existe différentes zones de stockage permettant la communication entre différentes TPR et entre les TPR et TDS. Ces zones de stockage permettent aux transactions de communiquer des données et d'accéder à ces dernières.

TDS-STORAGE	Zone de communication destinée à la transmission des informations entre TDS et les TPR.
TRANSACTION-STORAGE	Zone permettant à une TPR de transmettre des informations à la TPR suivante au sein d'une même transaction.
COMMON-STORAGE	Zone partagée par toutes les transactions d'une application TDS, permettant à une TPR de transmettre des informations à la TPR suivante de la même transaction ou d'autres transactions.
CONTROLLED COMMON-STORAGE	Zone de communication, gérée par le mécanisme GAC-EXTENDED, qui peut faire l'objet d'une restauration non actualisée en cas d'incident.
SHARED-STORAGE	Zones de stockage utilisateur partagée par deux ou plusieurs transactions.
CONSTANT-STORAGE	Zone contenant des caractères de contrôle de communications.

Création d'une application TDS

PRIVATE-STORAGE	Partie de la zone TRANSACTION-STORAGE attribuée à un utilisateur du début à la fin de la session.
WORKING-STORAGE	Zone de stockage facultative utilisée par une TPR comme dans un programme standard COBOL.

3.2.2.6 Affichage d'informations au terminal

Le programmeur TDS peut afficher les données à l'écran de l'une des façons suivantes :

- En mode normal (mode ligne) dans lequel les commandes sont introduites après le message de guidage READY ou à la suite de messages envoyés par une transaction.
- En mode formaté, dans lequel les données sont introduites au moyen de grilles d'écran ou de menus, à l'aide de l'utilitaire FORMS.
- A l'aide de IMAGEWorks (produit multimédia) sur un ordinateur individuel (PC).

De plus, la fonction Terminal Adapter permet au programmeur TDS de modifier un profil utilisateur. Cette fonction permet aussi d'envoyer des messages externes à une grille active ou à la ligne d'état (ligne en bas de l'écran). Pour disposer de cette fonction, le programmeur doit ajouter les procédures TA aux TPR, et l'administrateur doit modifier le programme de génération de TDS. Pour plus d'informations sur la fonction Terminal Adapter, consulter le *Guide de l'administrateur TDS* et le *Manuel du programmeur COBOL -TDS*.

Quel que soit le mode sélectionné, l'utilisateur peut émettre des commandes ou lancer des transactions à partir d'un terminal.

Pour émettre des commandes, l'utilisateur en introduit le nom après le message de guidage "READY". Les commandes accessibles à un utilisateur final concernent l'affichage de la liste des identificateurs de messages, le réaffichage du dernier message, ou la connexion à une autre application TDS. Les caractères utilisés pour exécuter ces opérations doivent être définis par l'administrateur TDS.

Lorsque les transactions sont en cours d'exécution, l'utilisateur peut communiquer uniquement avec la transaction. Lorsque celle-ci est achevée, il peut lancer une autre transaction ou une commande. Le terminal utilisateur est informé de la fin normale d'une transaction soit par l'émission du dernier message, soit par l'affichage du message de guidage "READY".

Mode normal

Le programmeur TDS prépare le mode normal en incluant les verbes SEND et RECEIVE dans les TPR. En mode normal, les utilisateurs finals peuvent introduire les commandes après "READY" (ou un autre message propre au site) ou après les messages émis par la transaction en cours d'exécution. (Le programmeur peut modifier le message de guidage "READY" suivant les spécifications du site.). Dans ce mode, les données sont introduites ligne par ligne.

Mode formaté

En mode formaté, une grille ou un menu apparaît à l'écran. Une grille est un ensemble de blocs de texte avec des zones vierges ou zones d'entrées dans lesquelles l'utilisateur final saisit les informations. L'utilitaire FORMS permet au programmeur TDS de créer et ajouter des grilles dans une application.

Une application TDS accepte les données introduites via les grilles, traite ces dernières, accède à la base de données, la met à jour, et produit des états comme si les données étaient saisies en mode normal. Une transaction active une grille, la traite, puis renvoie une grille mise à jour au terminal. La grille de sortie peut être soit une mise à jour de la grille d'entrée, soit une grille totalement nouvelle.

Lorsque les grilles sont utilisées dans une application, l'exactitude des données peut être vérifiée en les comparant aux paramètres autorisés. Il s'agit généralement des valeurs minimum et maximum autorisées pour la transaction concernée. Les grilles peuvent également délivrer des messages d'aide et d'erreur concernant chaque zone d'entrée (par exemple, pour indiquer l'introduction d'une valeur incorrecte).

La figure 3-8 présente une grille client.

ACHAT DE VALEURS

NUMERO DE CLIENT : #####

VALEUR	QUANTITE	DESCRIPTION	RIX	TOTAL
#####	#####	# # # # # #	# # #	####

CONFIRMATION: #

Figure 3-8. Exemple de grille

Création d'une application TDS

Mode fenêtrage IMAGEWorks

Sur un système disposant d'IMAGEWorks, l'utilisateur final peut accéder à des services multimédia à partir de l'application TDS. IMAGEWorks permet d'émuler un terminal TDS dans une fenêtre de PC. Une TPR peut ouvrir d'autres fenêtres via IMAGEWorks et afficher des listes ou des documents.

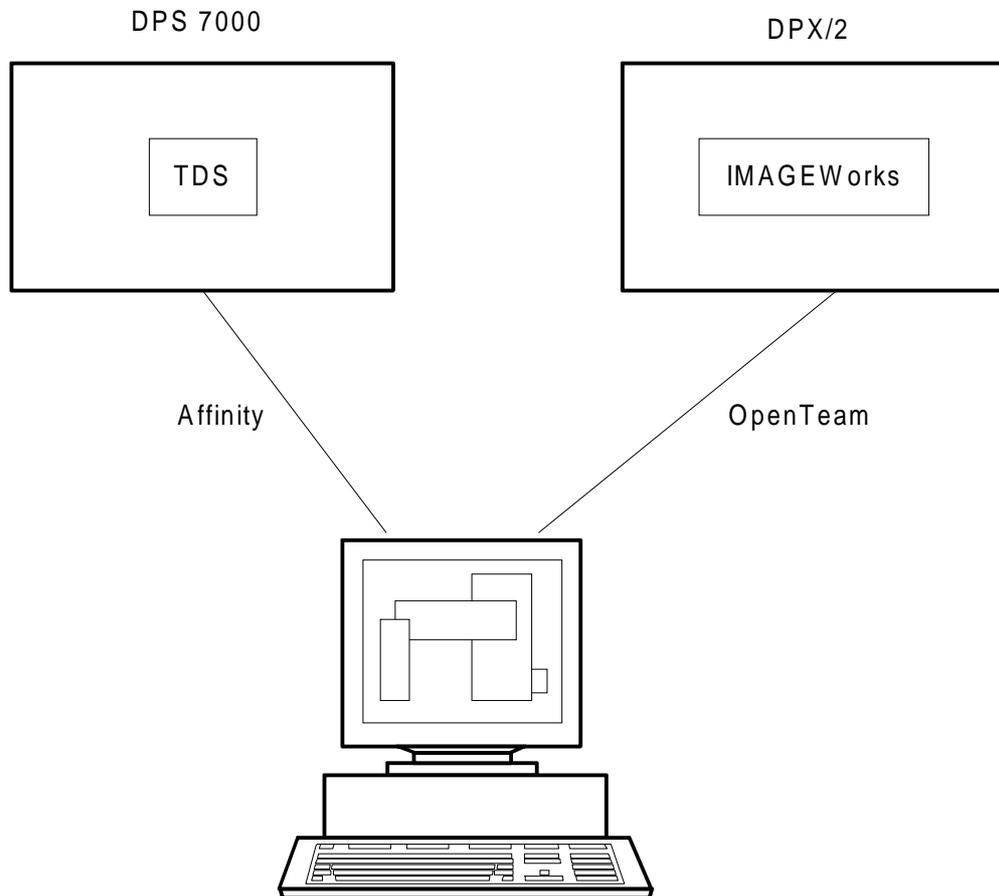


Figure 3-9. Utilisation de TDS avec IMAGEWorks

Avec IMAGEWorks, l'application peut s'afficher soit en mode normal soit en mode grille. IMAGEWorks permet à une application TDS de dialoguer avec un DPX, l'utilisateur final peut donc ainsi obtenir des informations de chaque système.

3.2.2.7 Gestion des états

L'éditeur de sorties sur terminaux GTWriter permet d'utiliser tout terminal imprimante du réseau comme imprimante : tout utilisateur TDS peut donc demander l'impression de sorties ou de fichiers sur n'importe quel terminal connu de GTWriter.

Pour disposer de cet utilitaire, le programmeur TDS doit ajouter les procédures GTWriter aux TPR. Pour plus d'informations, consulter le *Manuel du programmeur COBOL - TDS* ou le *Guide de l'utilisateur GTWriter*.

3.2.3 Assemblage des éléments de l'application

L'administrateur TDS réunit les différents éléments de TDS en générant d'abord le réseau, en exécutant l'utilitaire TP7PREP pour produire les fichiers TDS, puis en exécutant l'utilitaire TP7GEN pour créer le module chargeable TDS.

Le programmeur TDS compile les segments du code TPR origine (produisant les unités compilées), puis effectue l'édition des liens (voir figure 3-10).

A l'aide de l'utilitaire SYSMANT, l'administrateur TDS charge tous les modules partageables en mémoire de réserve.

Lorsque l'application TDS est en exécution et que les transactions sont lancées, les segments de code sont chargés en mémoire principale à partir de ces modules partageables en fonction des besoins. Autrement dit, TDS trouve la TPR adaptée dans la mémoire de réserve et l'introduit en mémoire principale pour l'exécuter.

Création d'une application TDS

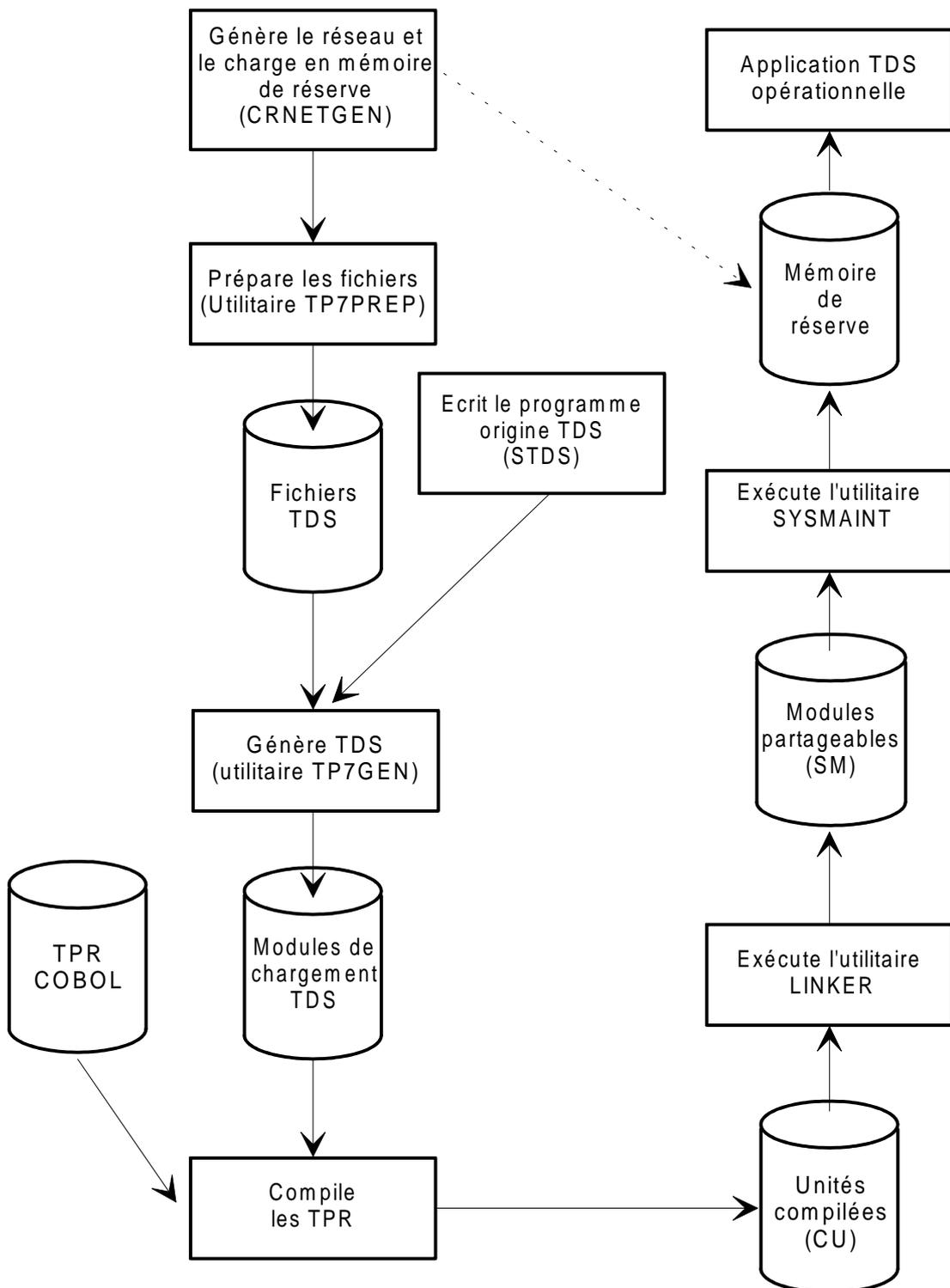


Figure 3-10. Assemblage de TDS

La figure 3-10 explique comment le TDS doit être assemblé. Les étapes à suivre sont indiquées dans l'ordre, cependant les étapes 4 et 5 peuvent être exécutées simultanément.

1. L'administrateur TDS ou l'administrateur du système génèrent le réseau à l'aide de l'utilitaire CRNETGEN.
2. L'administrateur TDS réserve de la place aux fichiers TDS au moyen de l'utilitaire TP7PREP.
3. L'administrateur TDS écrit le programme origine (STDS).
4. L'administrateur TDS exécute le programme de génération de TDS au moyen de l'utilitaire TP7GEN. Cet utilitaire produit les fichiers de commandes permettant l'édition de liens des TPR et le module chargeable TDS.
5. Le programmeur TDS écrit les TPR et les stocke dans une bibliothèque origine.
6. Le programmeur TDS compile les TPR origine. La bibliothèque d'entrée (<nom-tds>.COBOL) contient les définitions de fichiers et les zones de stockage décrites dans le programme de génération.
7. Le programmeur TDS stocke les unités compilées dans la bibliothèque CULIB.
8. L'éditeur de liens, à l'aide du fichier de commandes produit à la génération de TDS, relie les TPR.
9. L'éditeur de liens stocke les modules édités résultants dans une bibliothèque de modules partageables (SMLIB). Une même bibliothèque peut comporter plusieurs modules partageables de TPR.
10. L'administrateur TDS emploie l'utilitaire SYSMOINT pour charger les modules partageables de TPR en mémoire de réserve.

L'application TDS est prête à l'emploi.

3.2.4 Modifications à posteriori d'une application TDS

Une fois l'application TDS lancée, l'administrateur ou le programmeur peuvent avoir besoin de la modifier ou de la mettre à jour suite à des erreurs, des modifications de besoins, ou de nouveaux développements.

Une application TDS nécessite moins de modifications si elle est correctement mise en place. Selon la nature des modifications, le programmeur ou l'administrateur peuvent intervenir à deux niveaux :

- Les clauses du programme de génération de TDS peuvent être modifiées à tout moment. Dans un tel cas, l'utilitaire TP7GEN doit être exécuté de nouveau pour que les clauses soient prises en compte de façon permanente. Il faut savoir que la modification ou l'introduction d'une clause dans le programme de génération de TDS peut impliquer d'autres modifications. Toutes les TPR affectées par les nouvelles clauses doivent être de nouveau compilées, reliées par l'éditeur de liens et chargées en mémoire de réserve.

- Il est possible de modifier, d'effectuer l'édition de liens et de charger de nouveau les TPR sans exécuter le programme de génération de TDS (utilitaire TP7GEN).

3.3 EXECUTION D'UNE APPLICATION TDS

Une fois que l'administrateur et le programmeur ont créé l'application, l'opérateur maître peut lancer l'application TDS ; les utilisateurs finals peuvent ensuite s'y connecter.

3.3.1 Responsabilités de l'opérateur maître

L'opérateur maître est l'utilisateur chargé de contrôler une application TDS en exécution. Il est nécessaire que l'opérateur de terminal maître entre en communication avec l'application TDS pour que les utilisateurs finals puissent s'y connecter. En outre, l'opérateur maître est chargé des tâches ci-après :

- lancement de l'application TDS,
- ouverture et fermeture des fichiers,
- exécution des relances après incident,
- arrêt de l'application TDS.

Par ailleurs, l'opérateur maître dispose d'un ensemble de commandes particulières, appelées commandes maître, qui lui permettent de contrôler une application TDS active. Ces commandes permettent de modifier ou afficher des informations concernant l'application. Pour plus d'informations à ce sujet, consulter le *Guide de l'administrateur TDS*.

3.3.1.1 Lancement d'une application TDS

Une session est constituée par la durée d'une activité. Une session GCOS 7 débute au chargement du système (le message GCOS READY apparaît). La session de communications est lancée lorsque l'opérateur de la console principale introduit la commande requise. Ensuite, une session TDS peut être lancée par l'opérateur maître à partir de la console système ou de n'importe quel terminal. Jusqu'à 5000 sessions (utilisateur final) peuvent être connectées à une application TDS. Les trois sessions mentionnées précédemment sont toutes nécessaires à l'exécution d'une application transactionnelle.

La figure 3-11 présente la relation entre les différents types de sessions.

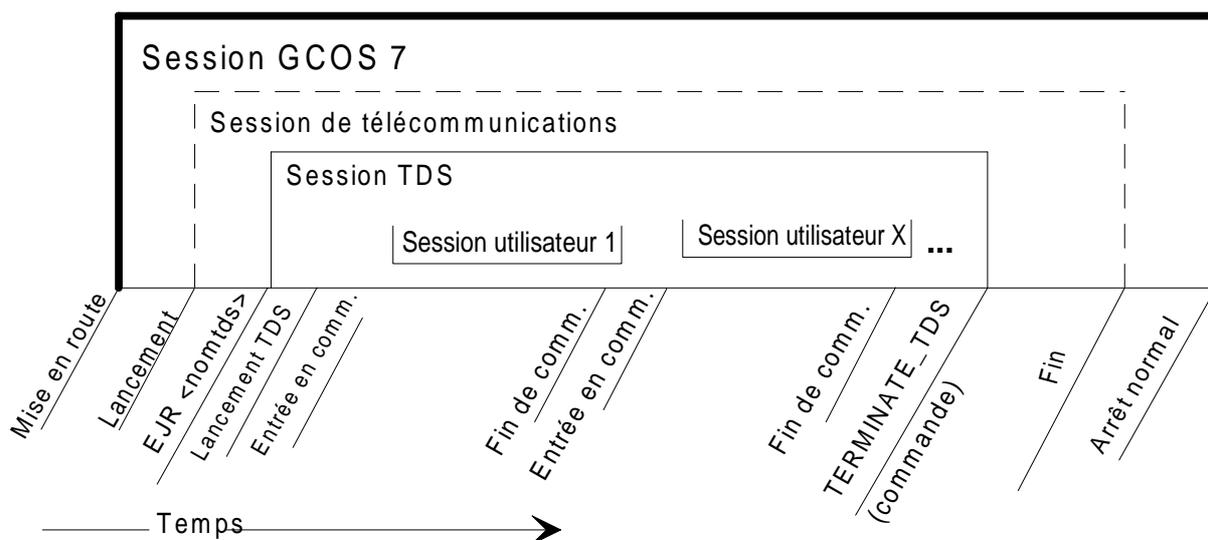


Figure 3-11. Différents types de sessions

Pour lancer une application TDS, l'opérateur maître introduit la commande GCL ENTER_JOB_REQUEST (EJR) en indiquant le nom de l'application TDS. Les ordres JCL permettant l'activation d'un travail TDS sont décrits dans le *Guide de l'administrateur TDS*.

Une fois la session TDS lancée, les opérations TDS sont sous le contrôle de l'opérateur maître, qui peut émettre des commandes maîtresses privilégiées. N'importe quel terminal du réseau peut agir en tant que maître si l'opérateur maître y est connecté. Tous les autres terminaux du réseau sont reconnus comme terminaux utilisateurs ; après être entrés en communication, ils peuvent lancer des transactions.

Le concept de boîte aux lettres est associé à celui de l'opérateur maître. Sous TDS, il s'agit d'une structure de données réservée à l'utilisateur qui gère la session TDS.

- Si la boîte aux lettres maîtresse est définie par l'administrateur TDS à la génération de TDS, **le premier terminal du réseau doté des droits d'accès appropriés pour se connecter** à cette boîte aux lettres maîtresse devient terminal maître. Aucun autre utilisateur ne peut se connecter à cette application TDS avant l'opérateur maître.
- Si aucune boîte aux lettres maîtresse n'a été définie par l'administrateur TDS, **l'utilisateur qui soumet le travail TDS** devient automatiquement opérateur maître.

Création d'une application TDS

Par exemple, lorsque le travail TDS est soumis à partir d'un terminal utilisateur quelconque et qu'aucune boîte aux lettres n'est définie par l'administrateur, le premier terminal qui se connecte devient opérateur maître, comme le montre la figure 3-12.

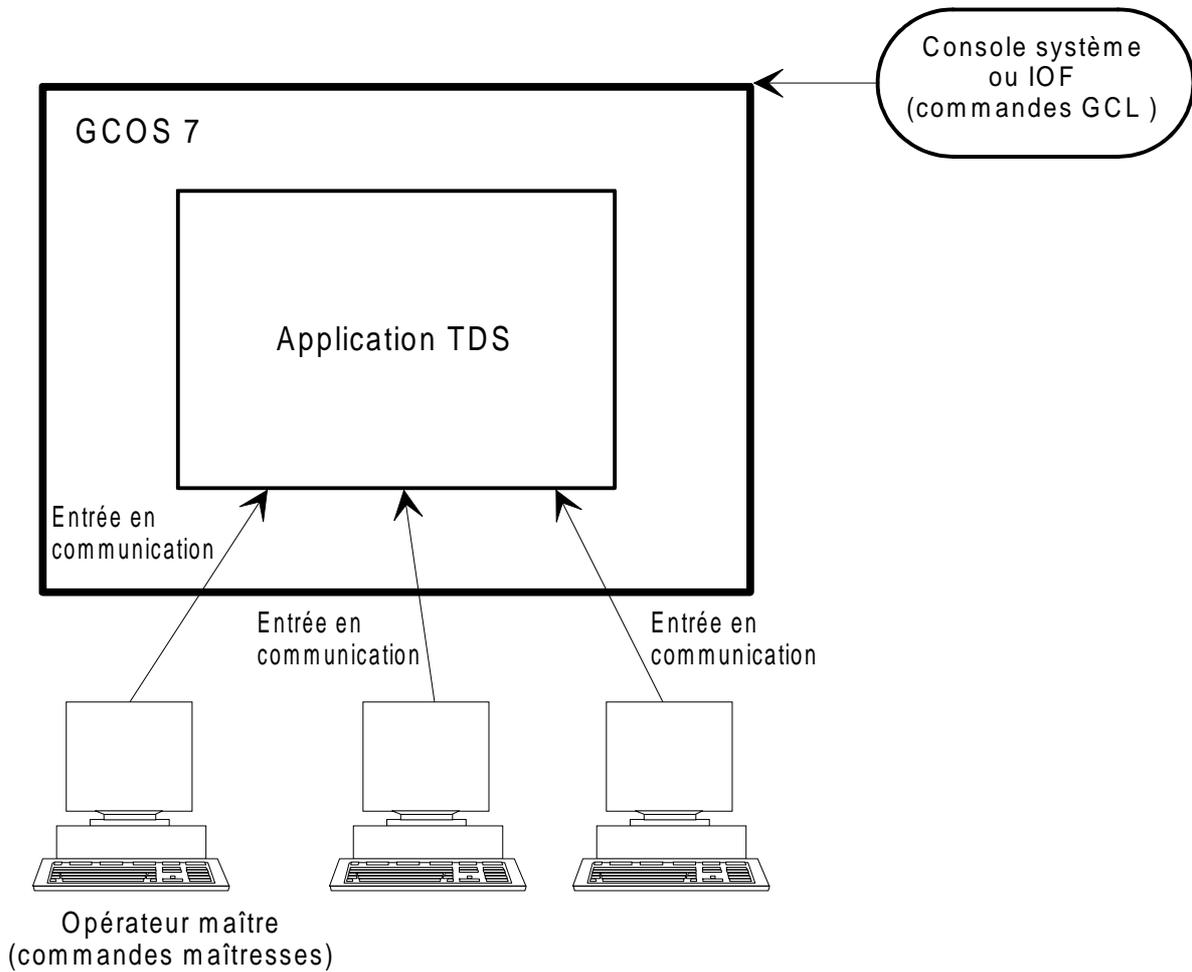


Figure 3-12. Terminal utilisateur final servant de terminal maître

Cependant, si le travail TDS est soumis à partir de la console système ou d'un terminal IOF, la console ou le terminal servent de boîte aux lettres maîtresse comme illustré par la figure 3-13.

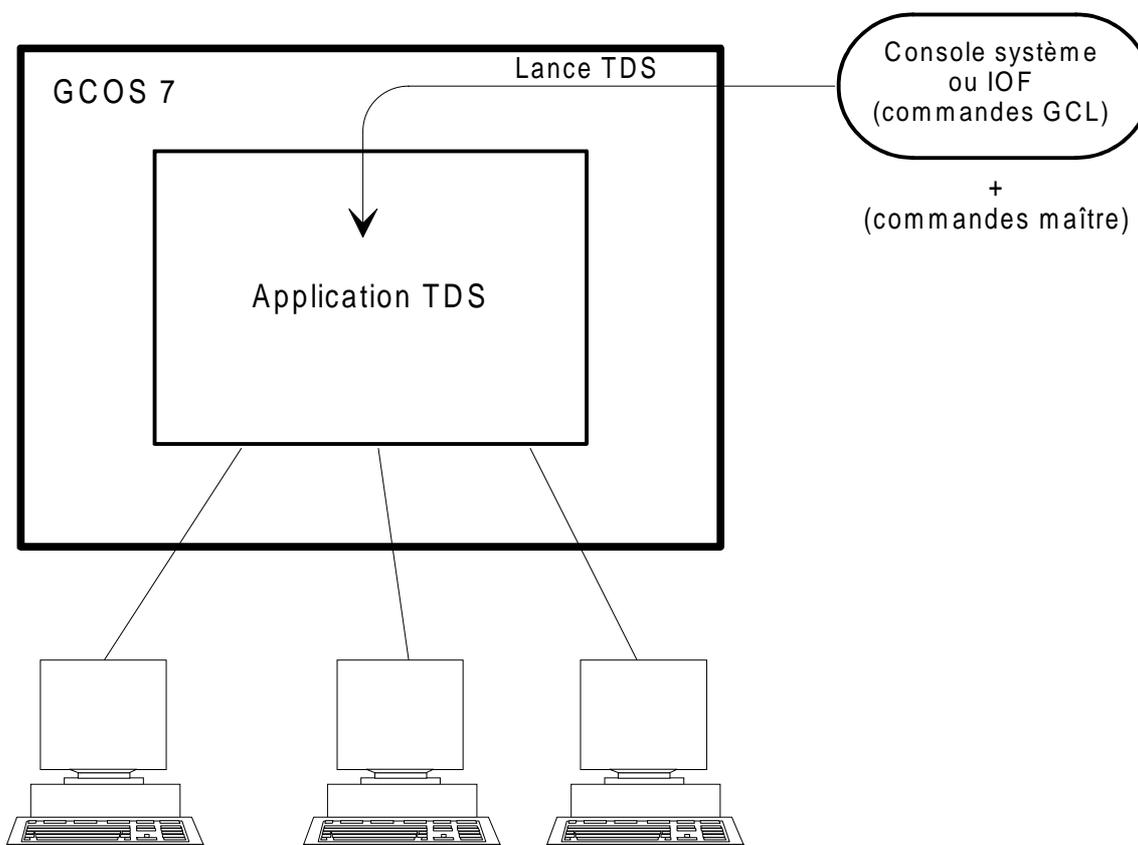


Figure 3-13. Console système ou terminal IOF servant de terminal maître

3.3.1.2 Utilisation de l'exploitation programmée pour gérer une application TDS

Un programme peut être écrit à l'aide de la fonction d'exploitation distribuée (DOF 7-PO). Elle permet à une application TDS d'être gérée automatiquement par un poste opérateur programmé. Celui-ci peut gérer simultanément plusieurs applications TDS.

Cette fonction gère les échanges, les réponses ou les messages non sollicités au moyen de services tels que la mise en file d'attente de données et la notification d'événements au destinataire. Pour plus d'informations concernant l'exploitation programmée, voir le *Guide de l'utilisateur DOF 7-PO*.

Création d'une application TDS

3.3.1.3 Ouverture et fermeture de fichiers

Après lancement d'une session TDS, tous les fichiers définis dans le programme de génération de TDS sont automatiquement ouverts. Cependant, l'opérateur maître peut les fermer ou les rouvrir en dynamique au moyen des commandes maîtresses. A la fin d'une session TDS, tous les fichiers sont fermés automatiquement.

3.3.1.4 Relance après incident

L'opérateur maître doit intervenir en cas d'incident empêchant l'utilisateur final d'exécuter normalement des transactions.

Si une session utilisateur s'interrompt prématurément sans que l'utilisateur se soit déconnecté (commande BYE), cet utilisateur est considéré comme "gelé" et ne peut introduire aucune commande. L'interruption peut être provoquée par les événements suivants :

- terminaux déconnectés,
- crash système,
- arrêt prématuré de TDS (dû à une défaillance du système ou à une commande de l'opérateur maître)
- incident sur le réseau.

L'opérateur maître doit déterminer la cause de l'incident puis relancer le TDS.

3.3.1.5 Arrêt d'une application TDS

L'opérateur maître peut se déconnecter sans affecter les utilisateurs finals qui restent connectés à l'application TDS. Cependant, l'opérateur maître peut arrêter une application TDS, au moyen de la commande `TERMINATE_TDS` en précisant si les autres utilisateurs doivent être déconnectés ou non.

En fonctionnement habituel, l'opérateur maître permet aux utilisateurs de se déconnecter à la fin de leurs transactions en cours. Dans ce cas, le message ci-après s'affiche aux terminaux des utilisateurs finals :

```
nom-tds SHUTDOWN  
      (nom-tds arrêt normal)
```

Si l'opérateur maître demande un arrêt immédiat (l'option `STRONG` est spécifiée dans la commande `TERMINATE_TDS`), toutes les transactions utilisateur en cours sont arrêtées prématurément.

3.3.2 Responsabilités de l'utilisateur final

La session utilisateur final commence lorsque l'opérateur de terminal se connecte à l'application TDS via un terminal. Cette session concerne toutes les activités menées par un utilisateur donné, y compris les réponses TDS, depuis la connexion de l'utilisateur à une application TDS particulière jusqu'à sa déconnexion. Un utilisateur final peut avoir plusieurs sessions au cours d'une seule session TDS.

3.3.2.1 Lancement d'une session utilisateur final

Pour se connecter à une application TDS, l'utilisateur introduit la ligne de commande suivante à partir de son terminal :

```
§*§ CN <nom d'une application TDS>
```

Lorsqu'un utilisateur se connecte à TDS, le système vérifie s'il y est autorisé ou non en contrôlant s'il appartient à un projet qui peut accéder à l'application. Si la connexion aboutit, le message de guidage "READY" (ou l'équivalent pour l'application) apparaît à l'écran. Lors de la demande de connexion, les utilisateurs inconnus, en doubles (c'est-à-dire déjà connectés) ou ceux fournissant des mots de passe incorrects sont rejetés. L'administrateur TDS peut également mettre en place une fonction permettant à un terminal d'être automatiquement connecté à une application TDS dès la connexion de l'opérateur de terminal à GCOS.

3.3.2.2 Lancement des transactions

Un utilisateur final connecté à une application TDS peut introduire un code de transaction (id-message) pour activer une transaction et en lancer l'exécution. L'id-message doit être défini par l'administrateur TDS lors de la préparation de l'application. Il peut comporter jusqu'à huit caractères, être suivi d'un espace, et inclure les paramètres demandés par la transaction.

Si l'id-message introduit est incorrect dans l'application TDS, le message "UNKNOWN TRANSACTION" ("*transaction inconnue*") s'affiche au terminal. Les utilisateurs finals peuvent activer uniquement les transactions que l'administrateur TDS leur a permis d'utiliser.

En mode transaction, l'utilisateur final peut interrompre une transaction en cours d'exécution en appuyant sur la touche BREAK ou en introduisant la commande `§*$BRK`. Ensuite, le déroulement des opérations dépend de la manière dont l'administrateur TDS a mis en place l'application TDS. Généralement, l'appui sur la touche BREAK en mode commande entraîne seulement l'émission d'un message "READY". Toutefois, si une commande BREAK est envoyée à partir d'un terminal passif, ce dernier est ramené à l'état actif. Pour plus d'informations sur les terminaux actifs et passifs, se référer au paragraphe "Correspondants" au chapitre "Conception et optimisation d'une application TDS".

3.3.2.3 Arrêt d'une session utilisateur final

A la fin de son exécution, une transaction envoie un état au terminal. L'utilisateur sait alors que la transaction est achevée, mais il reste connecté à l'application TDS. L'utilisateur peut soit lancer une autre transaction, soit se déconnecter du terminal.

Lorsque toutes ses activités TDS sont terminées, l'utilisateur final peut clore la session au moyen de la commande BYE. Avec cette commande, il est impossible que d'autres utilisateurs finals se connectent à la même session utilisateur. Par ailleurs, l'opérateur maître peut forcer un utilisateur final à se déconnecter.

La figure 3-14 illustre le déroulement d'une session utilisateur final.

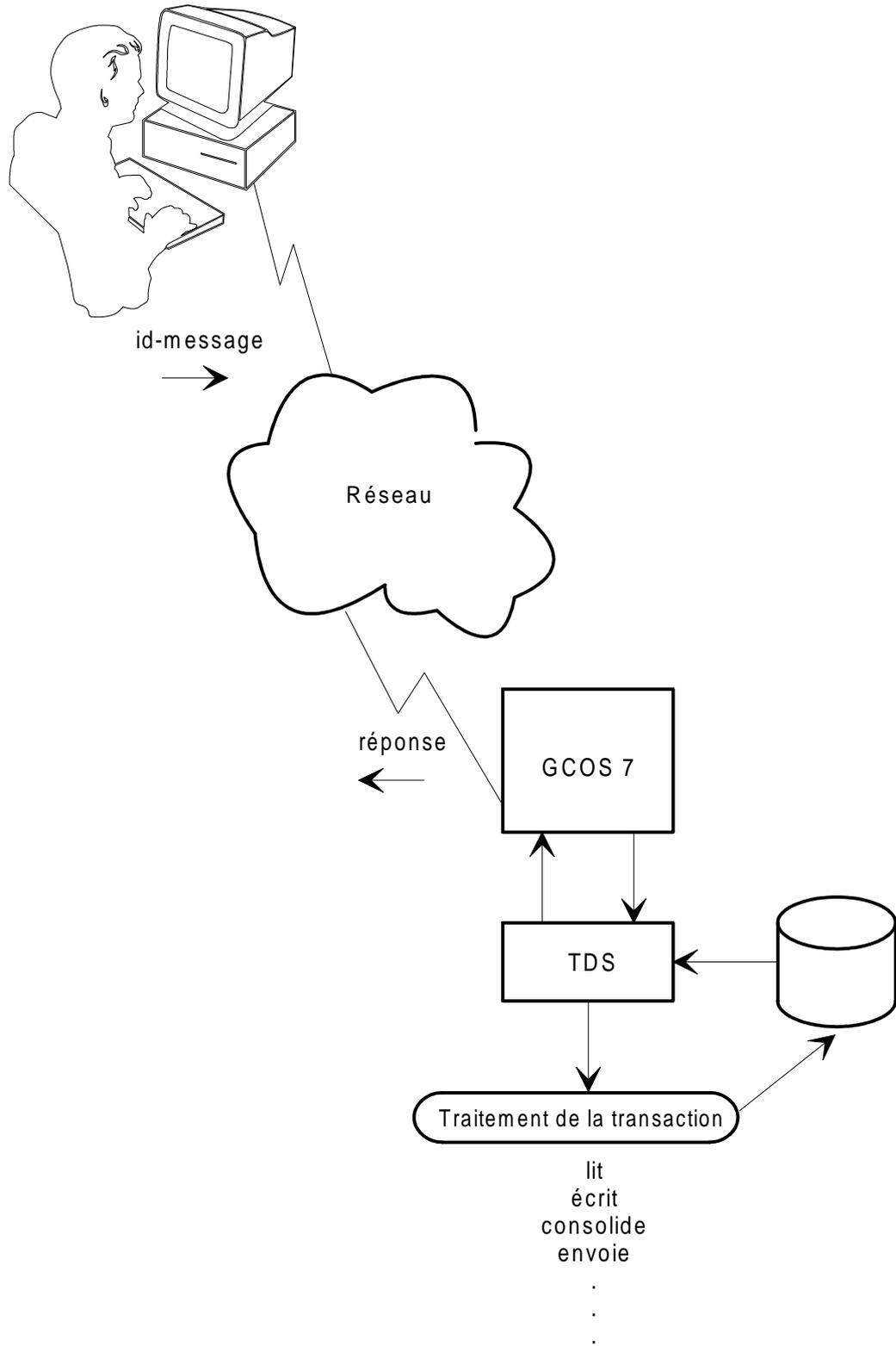


Figure 3-14. Déroulement de la soumission d'une transaction

4. Conception et optimisation d'une application TDS

Ce chapitre met l'accent sur les concepts que l'administrateur TDS doit maîtriser pour concevoir une application performante :

- simultanéité,
- non-concurrence,
- génération dynamique,
- affectation d'un process,
- correspondants,
- protocoles de communications,
- fonctions de sécurité.

4.1 CONTRÔLE DU TAUX ET DU VOLUME DE DONNÉES TRAITÉES

Sous GCOS 7-V6, une application TDS peut comprendre jusqu'à 2000 transactions. Les applications TDS utilisées pour les opérations bancaires, les réservations aériennes ou la gestion de stocks, doivent être capables de gérer un volume élevé de transactions. Dans des applications de grande taille, de nombreuses transactions s'exécutent simultanément.

Avec les applications pour lesquelles de nombreuses transactions s'exécutent simultanément et les fichiers sont accessibles à tous les utilisateurs, l'accès concurrent aux fichiers doit être contrôlé, afin d'assurer la cohérence logique des données. Lors de la génération de l'application, l'administrateur TDS assure le contrôle d'accès aux fichiers et aux ressources en spécifiant les clauses `SIMULTANEITY` (simultanéité), `NON-CONCURRENT` (non-concurrence) et `UNMAPPING` (libération d'un process).

- Le niveau de simultanéité détermine le nombre maximum de TPR exécutables simultanément.
- La non-concurrence empêche l'exécution simultanée des transactions. Elle permet d'éviter les conflits de ressources et d'interdire l'accès concurrent à des fichiers non contrôlés.
- La libération d'un process permet de remettre à disposition des ressources bloquées par les transactions en cours.

4.1.1 Transactions simultanées

L'administrateur TDS peut contrôler le nombre de transactions simultanées en spécifiant le niveau de simultanéité dans le programme de génération de TDS. L'administrateur TDS spécifie ainsi le nombre maximum de process disponibles pour le traitement des TPR, c'est-à-dire de définir le nombre maximum de TPR exécutables concurremment (124 sous V6).

Un niveau de simultanéité de 1 ne protège pas les fichiers des accès concurrents. Bien qu'une seule TPR puisse être exécutée à la fois, plusieurs transactions peuvent être actives en même temps, et leurs TPR respectives sont donc simultanées.

La figure 4-1 montre deux TPR et leurs transactions respectives essayant d'accéder à la même ressource.

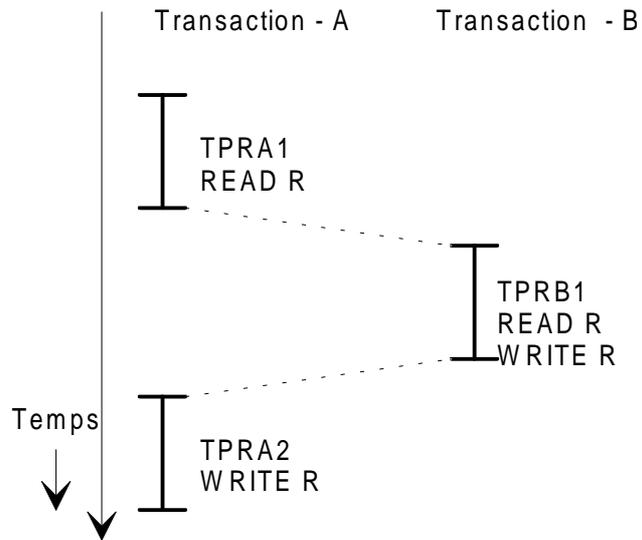


Figure 4-1. Niveau de simultanéité de 1

Malgré le niveau de simultanéité de 1, TPRA1 peut lire un article, puis TPRB1 peut lire et récrire ce même article avant qu'il soit écrit par TPRA2. Le fichier reflète uniquement la mise à jour exécutée par la transaction A.

Pour éviter ce type de problème, l'administrateur TDS peut spécifier la non-concurrence de certaines transactions, comme décrit ci-après.

4.1.2 Transactions non concurrentes

A la génération de TDS, l'administrateur TDS peut déclarer les transactions comme non concurrentes avec (une liste) d'autres transactions. Ce faisant, l'administrateur TDS :

- évite les conflits de ressources qui réduisent les performances du système,
- empêche les accès concurrents aux fichiers non contrôlés par TDS.

Dans une application TDS, beaucoup de tâches sont indépendantes et peuvent être exécutées en même temps (ou concurremment) par des transactions séparées. Par exemple, plusieurs utilisateurs peuvent concurremment lire ou mettre à jour des données dans une application TDS. Cependant, il est parfois avantageux d'effectuer les tâches les unes après les autres. Pour ce faire, il faut que les transactions soient déclarées non concurrentes, ou spécifier la non-concurrence.

La non-concurrence empêche l'exécution simultanée des unités de consolidation lorsqu'il y a un risque de conflit de ressources ou même d'interblocage. Elle permet à TDS de traiter ces conflits automatiquement, mais au détriment de la performance. Du point de vue de la performance, il est plus efficace de **ne pas** déclarer les fichiers non concurrents, mais d'utiliser des fichiers contrôlés par TDS et protégés par GAC-EXTENDED.

Le contrôle de la non-concurrence peut être automatique ou manuel selon le choix effectué lors de la génération de programme de TDS.

- Si le contrôle est automatique, TDS verrouille automatiquement les transactions non concurrentes au niveau de l'unité de consolidation. C'est-à-dire, dès qu'une transaction non-concurrente commence à s'exécuter, toute transaction avec laquelle elle est non-concurrente n'est pas autorisée à s'exécuter avant la fin de l'unité de consolidation.
- Si le contrôle est manuel, le programmeur TDS peut décider à son gré de la non-concurrence en déclenchant ou non la fonction de protection.

Il est nécessaire d'exécuter une transaction de façon non-concurrente avec une ou plusieurs autres transactions pour empêcher l'accès concurrent aux fichiers non contrôlés. Lorsque l'accès concurrent est autorisé pour des transactions (c'est-à-dire, lorsqu'elles **ne** sont **pas** déclarées non-concurrentes), le traitement du conflit des ressources diffère selon que le fichier est contrôlé ou non par TDS. Avec un fichier contrôlé par TDS, si deux TPR tentent de mettre à jour le même article, l'unité de consolidation peut être arrêtée prématurément. Dans le cas des fichiers non contrôlés, une des mises à jour peut ne pas être prise en compte.

La figure 4-2 montre la transaction-1 et la transaction-2 entrant en concurrence pour accéder à l'article-R. Alors que transaction-1 accède à l'article-R, elle contrôle complètement l'article. Pendant ce temps, la transaction-2 tente de contrôler ce même article. La transaction-2 est activée juste après la transaction-1, mais ne peut pas commencer à s'exécuter avant la fin de l'unité de consolidation de la transaction-1.

Si l'administrateur définit la non-concurrence, chaque unité de consolidation de la transaction s'exécute de façon non-concurrente. A savoir, l'article-R- est d'abord soumis à l'accès de l'unité de consolidation de la transaction 1 puis à celui de l'unité de consolidation de la transaction 2.

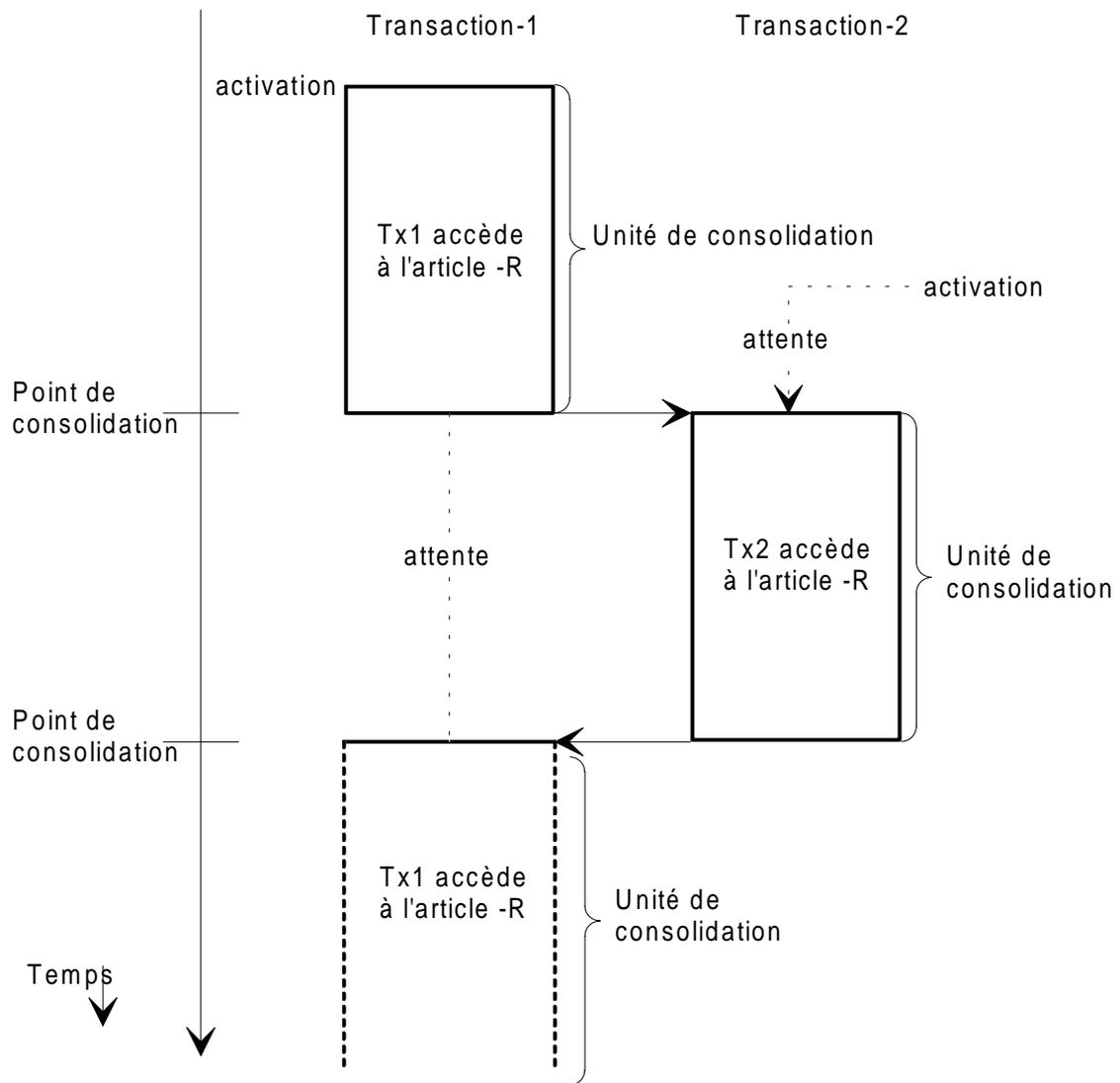


Figure 4-2. Transactions définies comme non-concurrentes

Même lorsqu'une unité de consolidation correspond à plusieurs TPR (figure 4-3), il n'y a aucun risque d'altération de fichiers, parce que l'ensemble du traitement exécuté par la première transaction est achevé avant que la seconde transaction puisse accéder à l'article.

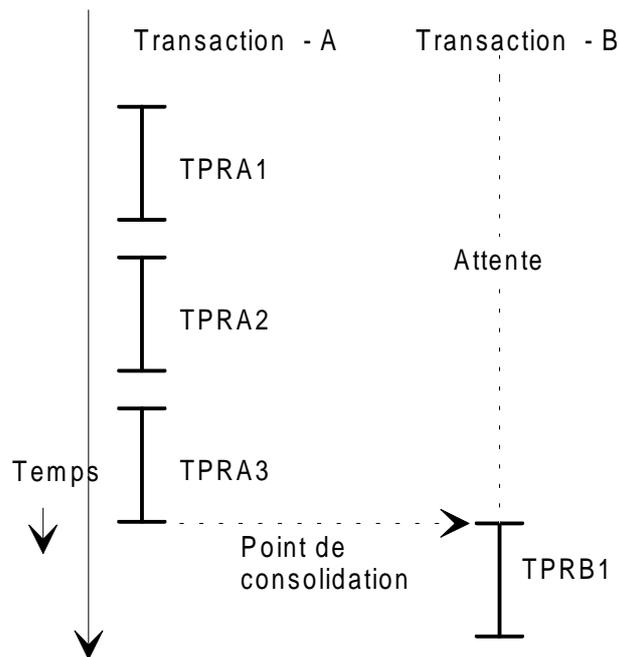


Figure 4-3. Unités de consolidation d'une transaction non-concurrente

Dans cet exemple, TPRA1 lit un fichier non contrôlé et TPRA2 y réinscrit les données mises à jour. Cette modification effectuée, TPRB1 peut lire et récrire le même fichier.

4.1.3 Affectation et libération de ressources

Dans une application TDS, il existe des tâches à vocation multiple partagées par les transactions. (Une tâche est un ensemble de ressources système nécessaires à l'exécution d'une séquence de code.) Ces tâches correspondent aux "process" physiques dans le sens de la terminologie GCOS7, et sont destinées à l'exécution des TPR.

Pour traiter une transaction, TDS affecte un de ces process à une TPR, puis exécute la TPR. L'affectation d'une tâche de gestion à une TPR est appelée **affectation d'un process** ("mapping"). A la fin d'une TPR, TDS détermine si une consolidation doit être effectuée et si le process est libéré. Lorsqu'il est **libérée**, la TPR l'est aussi ("unmapped") et ce process peut être utilisée par une autre TPR. A noter que l'affectation d'un process s'effectue au début d'une TPR, et sa libération à la fin.

Lorsqu'une TPR est libérée, son contexte de session (ou les informations relatives à la TPR et son état courant d'exécution) est sauvegardé de façon à pouvoir être reconstitué ultérieurement pour servir à la TPR suivante. Le contexte d'une TPR est sauvegardé dans un fichier SWAP. Si une TPR doit lire le contexte de la TPR précédente, elle accède au fichier SWAP pour le lire.

La figure 4-4 montre deux transactions. Dans la première transaction les TPR sont libérées, dans la seconde les ressources sont conservées ou déclarées par la clause "NO AUTOMATIC UNMAPPING" (libération de process non automatique).

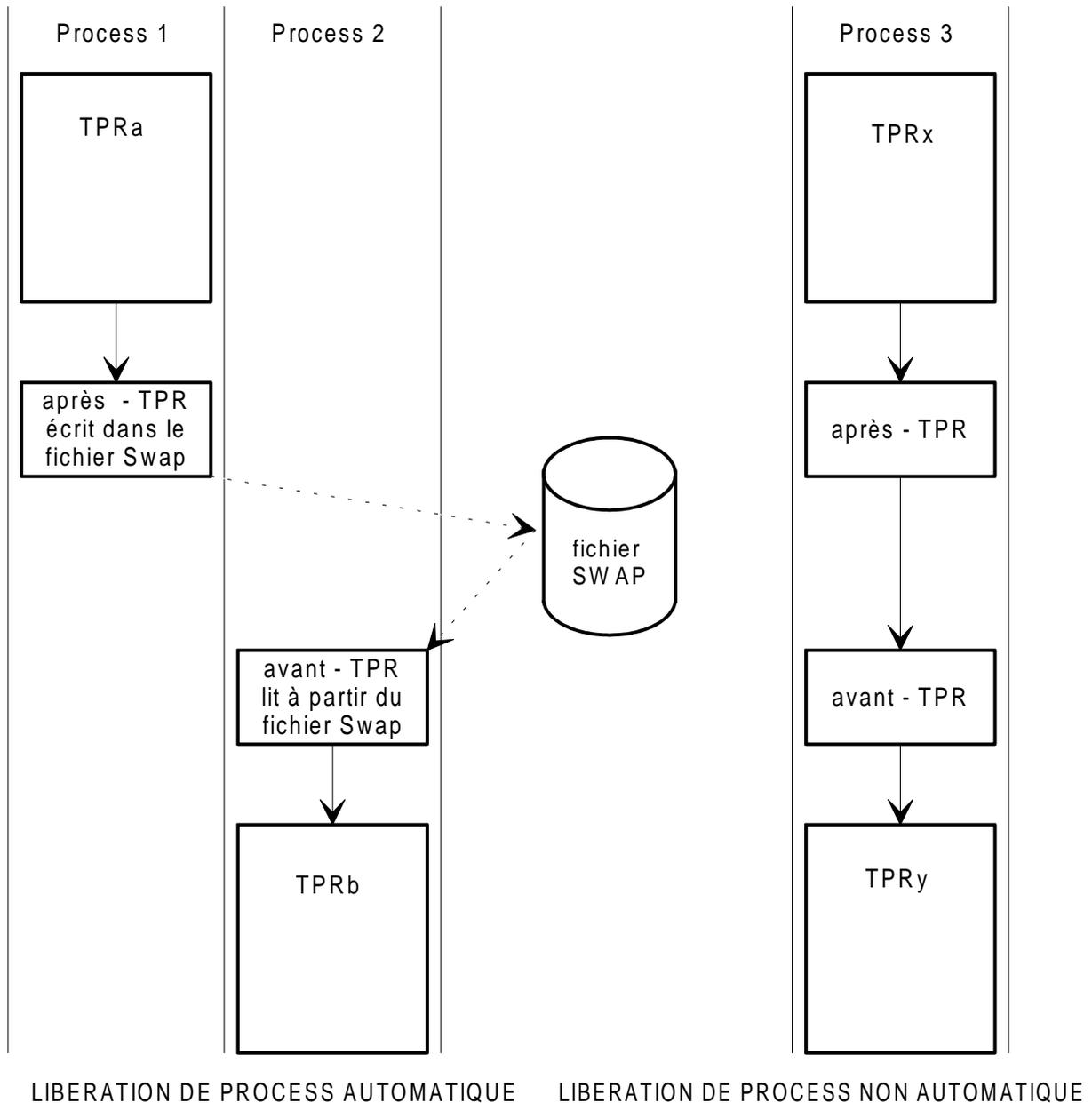


Figure 4-4. Libération de process automatique ou non automatique

L'utilisation du fichier SWAP pour la libération d'un process accroît le nombre d'opérations d'E/S. Cependant, lorsque les TPR sont conservées, le temps de réponse peut s'accroître en raison des files de TPR en attente de process.

Les transactions libérées n'occupent un process que pendant l'exécution d'une TPR. La TPR le libère lorsqu'une conversation est lancée. La libération d'un process permet à quelques tâches de gestion de servir un grand nombre de transactions, s'exécutant presque en parallèle.

L'administrateur TDS peut demander soit la libération systématique du process à la fin de chaque TPR, soit sa conservation. Si les TPR sont conservées, la charge d'E/S est réduite puisqu'il n'est pas nécessaire de sauvegarder le contexte de transaction entre les TPR. Ceci est particulièrement intéressant pour les transactions composées de nombreuses TPR. Bien que la libération de process non automatique (clause "NO AUTOMATIC UNMAPPING") améliore la performance de certaines transactions, elle peut réduire la performance globale d'un système.

4.2 COMMUNICATION ENTRE CORRESPONDANTS, APPLICATIONS ET AUTRES PRODUITS GCOS 7

Les applications TDS peuvent échanger des informations avec les applications et les utilisateurs d'autres applications TDS.

Ce paragraphe décrit :

- les types de correspondants,
- la génération dynamique,
- les protocoles de communication destinés au dialogue au sein d'une application TDS et/ou entre TDS et IBM, ou avec les applications compatibles IBM, ainsi que les protocoles destinés aux transactions exécutées sous UNIX.

4.2.1 Correspondants

Définition d'un correspondant

Un **correspondant** est simplement un "objet" avec lequel une application peut "dialoguer". Il peut s'agir d'un terminal (ou de l'utilisateur final qui y est connecté), d'un correspondant "virtuel", ou même d'une application. Les correspondants doivent être définis dans le programme de génération de TDS en vue d'établir les connexions nécessaires. Cependant, les correspondants qui demandent une connexion à TDS, par exemple un terminal, n'ont pas besoin d'être déclarés dans TDSGEN.

Correspondant terminal

Généralement, les correspondants de type terminal sont reliés à TDS lors de leur entrée en communication. Toutefois, certains correspondants tels que les imprimantes peuvent être connectés automatiquement. Les applications sont connectées à TDS à la demande de l'opérateur maître.

L'opérateur maître est le correspondant responsable de l'application TDS. Comme tout autre utilisateur, il peut lancer des transactions.

Les correspondants de type terminal peuvent fonctionner en mode actif ou passif.

- Un terminal en **mode actif** peut lancer des transactions et disposer d'un grand nombre de fonctions.
- Un terminal en **mode passif** ne peut pas lancer des transactions. Il peut recevoir des messages et des transactions peuvent être activées à distance pour son compte. Ce type de terminal peut participer à des conversations, mais il n'est pas mis en mode saisie lorsque la transaction prend fin.

Un terminal passif est souvent un terminal sans clavier, mais il peut également être rendu passif par le programmeur TDS. Le programmeur peut rendre passif un terminal avec clavier au moyen d'une TPR spécifique.

Correspondant application

L'accès à un correspondant de type application requiert l'utilisation d'un protocole de communication (voir le paragraphe "Protocoles de communication entre correspondants" dans ce chapitre).

4.2.2 Génération dynamique de tâches à partir d'une transaction (SPAWN)

Le lancement d'une nouvelle transaction à partir d'une transaction est appelée **génération dynamique**. De cette manière, un correspondant lance une transaction pour le compte d'un autre correspondant, à partir d'une transaction active. La transaction ainsi lancée est dite transaction générée dynamiquement, par opposition à celle demandant la génération dynamique.

Lorsqu'une transaction est générée dynamiquement, son exécution est indépendante de la transaction qui a demandé la génération dynamique. Les transactions générées dynamiquement sont mises en files d'attente et lancées lorsque le correspondant requis par la transaction devient inactif. Une transaction générée dynamiquement est validée lorsque l'unité de consolidation dans laquelle s'exécute la génération dynamique se termine normalement.

La génération dynamique peut être employée lorsqu'une transaction a besoin d'accéder à des ressources (imprimantes notamment) que la transaction ne peut pas lancer. De même, elle peut provoquer le lancement différé d'une transaction à un moment donné, ou celui d'une transaction dotée de priorités différentes de celles définies initialement. Une transaction générée dynamiquement vers un correspondant virtuel peut uniquement interroger des paramètres transactionnels, elle ne peut pas lui envoyer d'informations.

La figure 4-5 illustre un exemple simple de génération dynamique. La transaction A demande la génération dynamique de la transaction B pour l'utilisateur X. La transaction B peut être exécutée immédiatement, ou après un délai spécifié, ou bien à une heure spécifiée, selon la demande de la transaction A.

L'instruction CALL "SPAWN" est exécutée en t2, mais n'est validée qu'en t3. La transaction B reste en file d'attente de génération dynamique de t3 à t10. Pendant ce temps, la transaction 0 (TX0) est exécutée pour l'utilisateur X. Puis en t10, la transaction B est exécutée pour ce même utilisateur.

Le lancement de la transaction B n'est pas fonction de la fin de la transaction A, sauf si le correspondant de la transaction A est le même que celui associé à la transaction B.

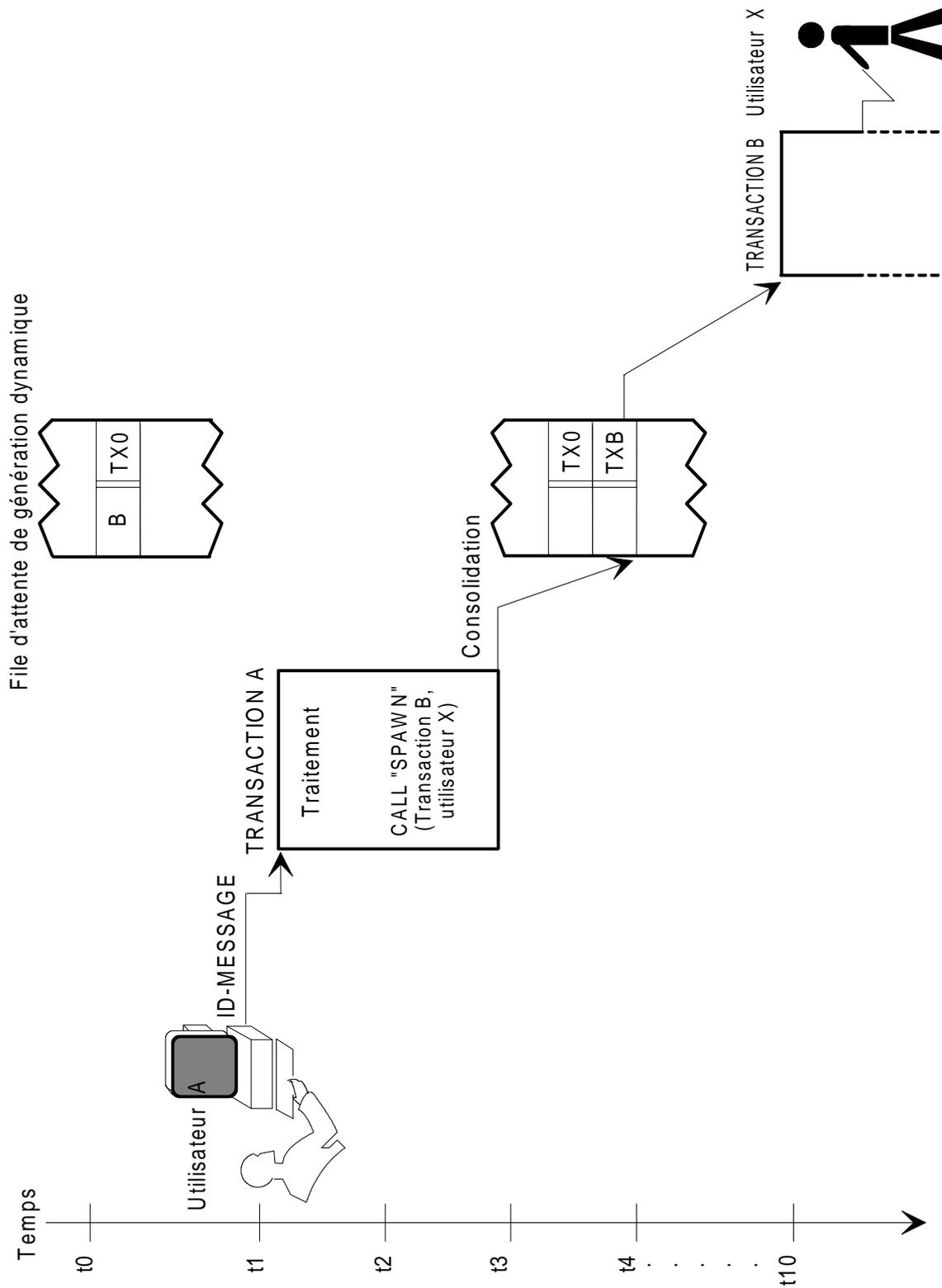


Figure 4-5. Génération dynamique

4.2.3 Protocoles de communication entre correspondants

Les utilisateurs communiquent avec TDS ou avec une application au cours d'une **session de communication**, c'est-à-dire d'un dialogue structuré. Les applications peuvent aussi communiquer entre elles. Une structure prédéfinie permet de normaliser ces dialogues. Cette normalisation est assurée par l'utilisation de **protocoles** qui constituent l'ensemble des règles de communication entre deux correspondants d'une session de communication.

TDS dispose des protocoles de communication ci-après :

- le protocole TM (Terminal Manager),
- la fonction de déport (PT),
- le protocole XCP1 (Extended Communication Protocol),
- le protocole XCP2 (Extended Communication Protocol).

4.2.3.1 Protocole TM

Ce protocole permet de gérer la communication entre un terminal ou groupe de terminaux et un correspondant. (Ce dernier peut être soit un autre terminal soit une application avec laquelle la session est établie.) Ce protocole traduit les données de l'application dans le format défini pour le terminal.

Il doit exister une connexion logique entre le protocole TM et le correspondant. Les informations nécessaires à cette connexion sont déclarées par l'administrateur lors de la génération du réseau.

4.2.3.2 TCAM et fonction de déport (PT) de TDS

La fonction de déport (PT) de TDS permet à un utilisateur final TDS de se connecter à des applications à distance sans devoir clôturer la session hôte TDS active. (La **session hôte** est l'application TDS initiale à laquelle l'utilisateur est connecté.) L'application à distance peut être une autre application TDS (sur le même site ou sur un autre site), ou IOF.

TCAM ouvre la session de déport TDS, ou liaison logique, entre l'utilisateur et l'application. TDS crée cette liaison au moment de la connexion. La figure 4-6 montre un utilisateur avec deux sessions : la session principale et la session de déport TDS. Une seule session de déport TDS peut être ouverte à la fois.

Après émission de la commande de déport TDS, l'utilisateur commence à travailler directement sous la ou les applications à distance auxquelles il est connecté, et non sous l'application TDS initiale. Pour revenir à l'application TDS initiale, il doit se déconnecter de l'application TDS à distance.

La figure 4-6 présente une session de déport TDS.

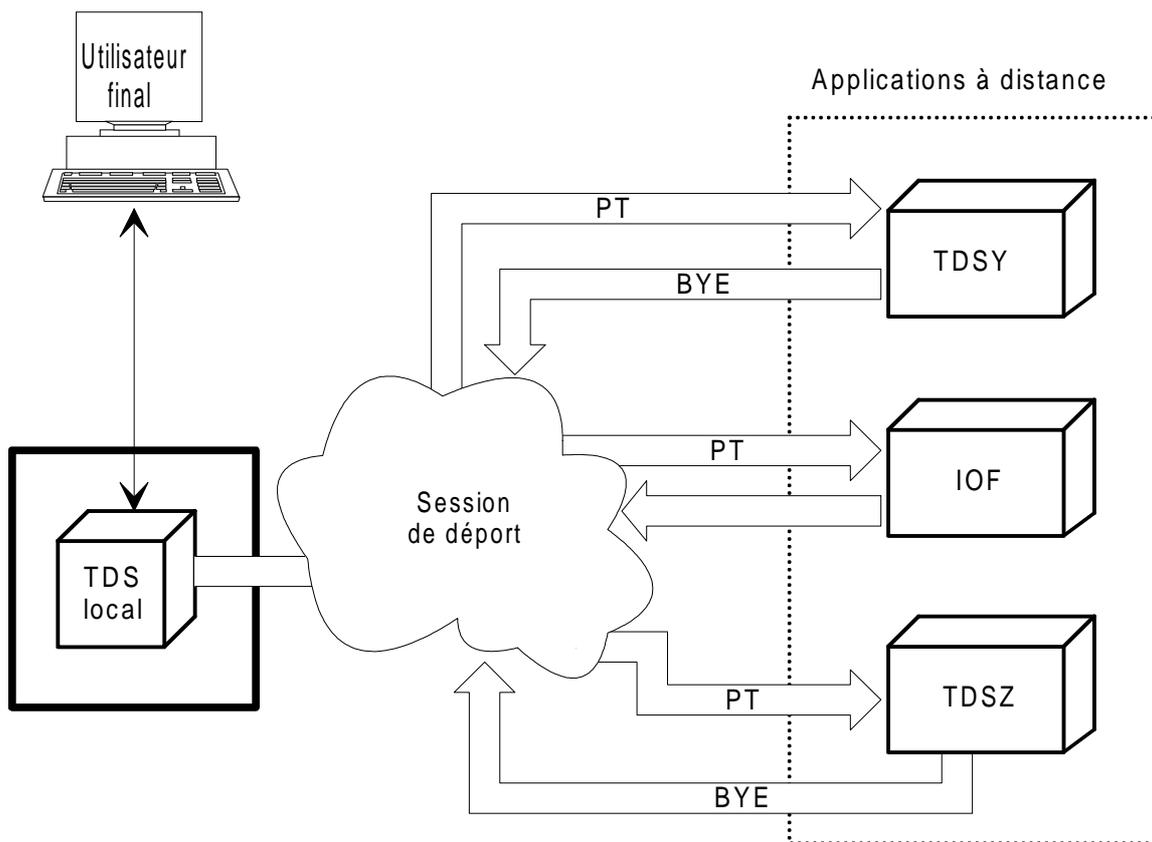


Figure 4-6. Connexion à des applications à distance avec la fonction de déport

4.2.3.3 Protocole XCP1

La famille des protocoles XCP permet aux utilisateurs de se connecter à d'autres applications pour effectuer des interrogations et des modifications. Les applications ne doivent pas forcément s'exécuter sous GCOS 7, mais sous n'importe quel système d'exploitation utilisant XCP1.

XCP1 offre une compatibilité avec les protocoles IBM LU0 (CICS) et LUP (IMS). Il permet aux applications d'échanger des messages avec ces autres applications. TDS met en oeuvre XCP1 et peut donc communiquer avec d'autres applications qui l'acceptent.

XCP1 permet aux données de circuler entre des applications qui s'exécutent sur des systèmes locaux ou à distance. ("A distance" signifie soit sur le même site, soit sur un autre site). La coopération a lieu au niveau de l'application, et non celui du correspondant. En d'autres termes, un correspondant peut exécuter plusieurs applications communiquant par le biais du protocole XCP1.

Lorsque TDS dialogue avec une application compatible IBM, il fonctionne comme application frontale, et IBM comme application back-end. En effet, une transaction GCOS 7 peut lancer l'application IBM, mais cette dernière ne peut pas lancer une application GCOS 7.

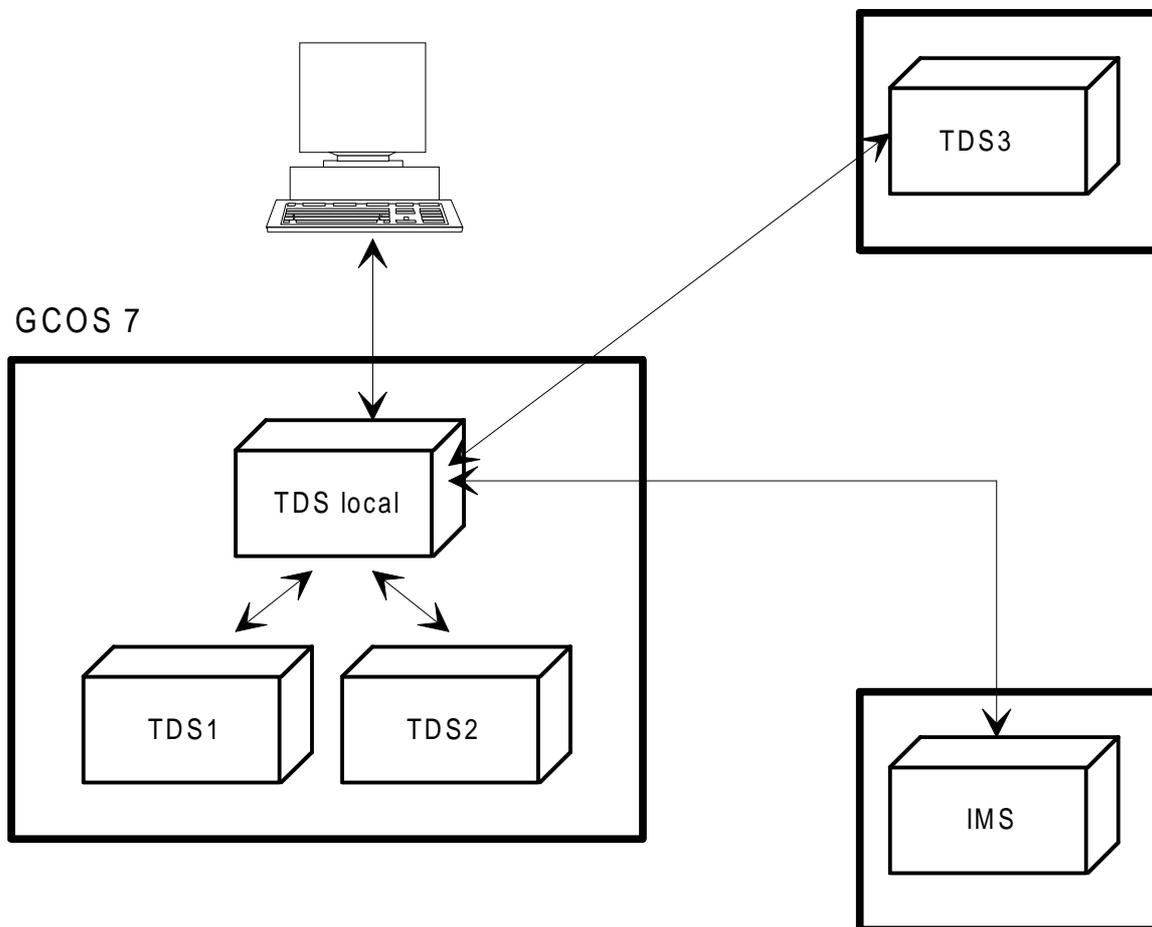


Figure 4-7. Session de communication via XCP1

La figure 4-7 montre comment une application TDS communique avec une autre application au cours d'une session de communication via le protocole XCP1 . Une transaction lancée par l'utilisateur final, puis exécutée dans l'application TDS locale de celui-ci, travaille avec des transactions dans quatre applications différentes :

- application TDS1,
- application TDS2,
- application TDS3 (sur un site à distance),
- application IMS (en environnement IBM).

Lorsque deux correspondants dialoguent au cours d'une session XCP1, l'un est dit primaire et l'autre secondaire.

Pour utiliser le protocole XCP1 dans une application TDS, le programmeur TDS doit concevoir des transactions avec les instructions CALL XCP1. Il doit modifier l'environnement TDS lors de la préparation de l'application. Pour plus d'informations sur TDS et XCP1, consulter le manuel *Intercommunication transactionnelle sous protocole XCP1*.

4.2.3.4 Le protocole XCP2

Le protocole XCP2 appartient à la famille des protocoles XCP qui permet la communication entre les applications.

Dans le cas de XCP2, cette communication permet la gestion de transactions distribuées et l'échange de messages.

Avec le protocole XCP2, une application TDS peut dialoguer avec :

- une autre application TDS du même système ou d'un autre système,
- une application IBM utilisant l'unité logique de type 6.2 de l'architecture SNA d'IBM,
- toute plate-forme sur laquelle existe un sous-système de type transactionnel utilisant un protocole XCP2 (par exemple, un système Bull DPX avec le produit CPI/C-OSI).

La figure 4-8 présente une session de communication via XCP2.

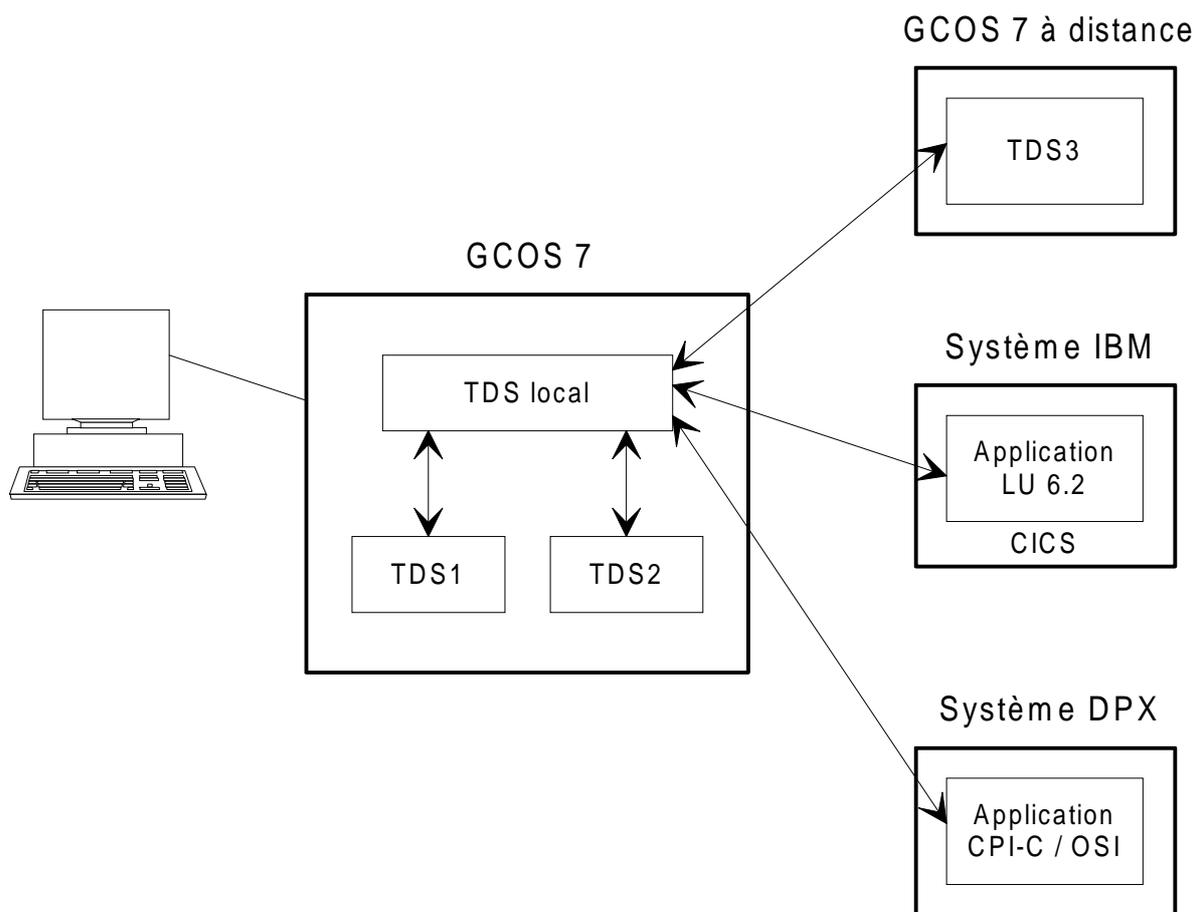


Figure 4-8. Session de communication via XCP2

Dans les transactions, le programmeur TDS utilise soit l'interface de programmation X/Open CPI-C, soit l'interface de programmation PPC-PI de Bull. L'administrateur TDS doit configurer un environnement TDS spécifique pour CPI-C/XCP2. Pour plus d'informations sur XCP2, voir le *Guide de l'utilisateur CPI-C/XCP2*.

XCP2 offre l'avantage de permettre à chacun des partenaires (applications) de lancer une transaction à distance.

4.3 DIALOGUE AVEC LE MONDE UNIX

2LTP (Two Level Transaction Processing)

Il est possible de se connecter à une application TDS à partir d'une application MS-DOS ou d'une station de travail UNIX. TDS considère l'application comme un terminal. Voir la documentation concernant Affinity et OTM.

CTP (Cooperative Transaction Protocol)

L'interface de programmation API CPI-C accepte le protocole XCP2 (au niveau "confirm") et permet la communication entre une application TDS et une application BOS/TP. Voir le *Guide du programmeur CPI-C*.

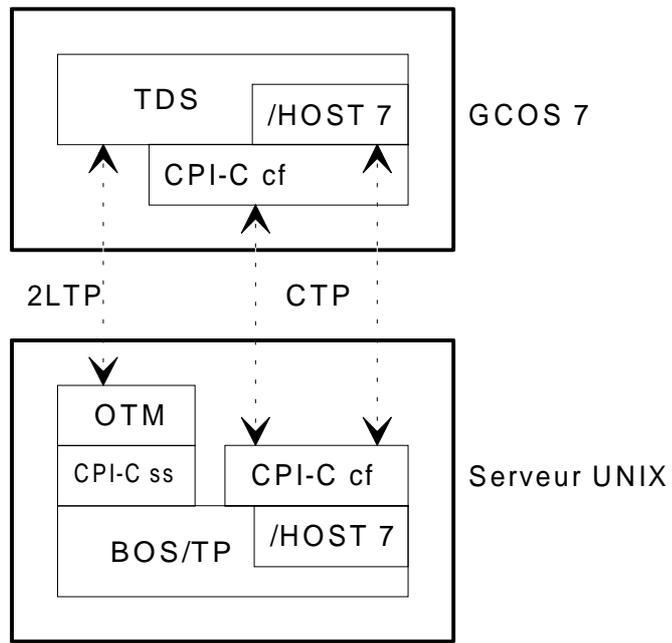
/HOST 7 est une fonction fondée sur CPI-C qui permet d'appeler une transaction TDS via l'interface API BOS/TP normale. Cette fonction est disponible uniquement à compter de l'état technique 6152.

IMAGEWorks

Il existe une interface API qui permet à une transaction TDS de manipuler un document IMAGEWorks. Voir le *Guide de l'utilisateur Liaison TDS-IMAGEWorks*.

La figure 4-9 présente TDS en environnement UNIX.

TDS - Concepts



- CTP = Co-operative Transaction Processing (traitement transactionnel coopératif)
- 2LTP = Two Level Transaction Processing (traitement transactionnel à deux niveaux)
- cf = Confirm Set } Interface de programmation API
- ss = Starter Set }
- OTM = Open Terminal Manager

Figure 4-9. TDS en environnement UNIX

4.4 ACCES ET ORGANISATION DES DONNEES DANS TDS

Une application TDS peut accéder simultanément à des données ORACLE, à des bases de données IDS/II, et à des fichiers UFAS. Une TPR peut mettre à jour soit une base de données ORACLE, soit des données IDS/II et UFAS.

Les données utilisées dans une application TDS peuvent être issues de l'une des bases de données ci-après :

- IDS/II
- IQS
- ORACLE

4.4.1 IDS/II et TDS

Les transactions de TDS peuvent accéder aux bases de données IDS/II. Pour ce faire, les aires IDS/II doivent être déclarées dans la génération TDS origine. Les schémas et sous-schémas peuvent être utilisés. Les aires IDS/II sont des fichiers UFAS contrôlés et suivent les règles d'intégrité habituelles. Voir le paragraphe "Fichiers contrôlés par TDS" dans le chapitre 2 et également la documentation IDS/II.

4.4.2 Utilisation d'IQS avec TDS

Les requêtes IQS peuvent s'exécuter dans un environnement TDS. Elles peuvent accéder aux bases de données IDS/II et UFAS-EXTENDED pour lesquelles des schémas et/ou vues ont été définis.

Sous TDS, les requêtes IQS peuvent utiliser des macros, des grilles, des formats de présentation et des descriptions d'état, exactement comme sous IOF. Ces objets peuvent être chargés explicitement par l'opérateur maître, pour servir à tous les utilisateurs, ou implicitement par IQS lorsqu'ils sont destinés à une requête (et sont donc réservés à l'utilisateur).

Lorsque les requêtes IQS sont compilées, elles sont reliées pour créer une TPR. Une transaction peut combiner des TPR COBOL et des requêtes IQS. Cependant, aucun transfert de données ne peut être effectué entre les TPR et les requêtes.

Une fois les requêtes écrites par le programmeur TDS, l'administrateur TDS doit déclarer les vues, schémas, fichiers et aires IQS dans le programme de génération de TDS. De plus, l'administrateur doit spécifier le fichier <nom-tds>.GMEM pour la préparation de TDS. Pour plus d'informations à ce sujet, voir le *Guide de l'utilisateur IQS-V4/TDS*.

4.4.3 ORACLE et TDS

Les bases de données ORACLE peuvent être utilisées avec TDS. Plusieurs applications TDS peuvent partager une base de données ORACLE et y accéder simultanément. La base de données peut se trouver sur le même système (en local) ou sur un système GCOS 7 à distance.

Sous ORACLE/TDS, le programmeur TDS écrit des TPR en COBOL comportant des instructions SQL imbriquées qui accèdent aux bases de données ORACLE. Les TPR écrites en langage C ne sont pas acceptées.

Le serveur ORACLE s'exécute indépendamment de l'application TDS. Les bases de données ORACLE disposent de leur propre fichier journal et de leurs propres segments de restauration non actualisée, et toute la maintenance est prise en charge par les outils ORACLE.

L'administrateur TDS doit déclarer la base de données dans le programme de génération de TDS. Pour plus d'informations à ce sujet, consulter le *Guide de l'utilisateur - ORACLE-V6/TDS*.

4.5 INTEGRITÉ & SECURITE : AUTRES PRODUITS UTILISES AVEC TDS

Différents produits de GCOS 7, fournis en option, peuvent être utilisés dans l'environnement TDS pour améliorer la sécurité du système. Ils permettent de réduire les effets d'incidents matériels et logiciels ou de tentatives d'accès non autorisés. Ces options GCOS 7 sont :

- TDS-HA,
- Fonction Disques miroirs (MIRROR),
- RDDF7,
- SECUR'ACCESS.

4.5.1 TDS-HA

L'objectif de la haute disponibilité (HA) est d'accroître le temps de disponibilité des applications, et de réduire l'impact des incidents sur les applications. TDS-HA est le produit GCOS 7 qui offre aux utilisateurs ce service en continu. TDS-HA accroît la disponibilité des applications TDS en cas d'incidents matériels ou logiciels. TDS-HA peut être mis en oeuvre sur une ou plusieurs applications TDS sur un site (avec 2 systèmes), et peut donc co-exister avec des applications non-HA telles que TDS ou IOF. TDS-HA peut utiliser les bases de données ORACLE et UFAS/IDS/II.

Le produit TDS-HA :

- réduit le temps de relance après un crash,
- relance l'ensemble des transactions à partir du dernier point de consolidation,
- fournit un système de secours en cas d'incident,
- exécute simultanément la reprise et reconnexion du système,
- reconnecte tous les utilisateurs connectés avant l'incident,
- est fondé sur des systèmes couplés,
- accepte la fonction Disques miroirs,
- reconstitue les fichiers de journalisation.

Pendant l'exploitation, l'application TDS est suivie par un TDS de secours fonctionnant avec un processeur distinct. En cas d'incident, le système bascule le traitement du TDS actif vers le TDS de secours. Le TDS de secours prend le relais de la session et continue normalement le traitement.

Pour plus d'informations sur la haute disponibilité, voir le *Guide de l'administrateur - Haute Disponibilité (HA)*.

4.5.2 Disques miroirs

La fonction Disques miroirs est un produit GCOS 7 qui offre la haute disponibilité des données enregistrées sur disques. Ce faisant, elle assure la disponibilité continue des données nécessaires aux applications.

La fonction Disques miroirs permet cette disponibilité en conservant les données en double, chaque copie étant enregistrée sur un disque distinct. A chaque disque correspond son image miroir, ou double. Chaque opération d'écriture génère deux écritures, une pour chaque double. Si l'un des doubles se perd, l'autre est disponible. En cas d'incident affectant les disques ou contrôleurs, la fonction Disques miroirs permet à l'application de continuer le traitement normal en utilisant le double des données.

La fonction Disques miroirs peut être utilisée sur des mono-systèmes, sur des systèmes couplés et sur des systèmes exploitant TDS-HA. Elle peut s'appliquer à des disques spécifiés ou à tous les disques d'un système. Pour cette fonction, il faut disposer de disques supplémentaires ; en effet, chaque disque déclaré comme disque miroir doit disposer d'un double de secours.

Pour plus d'informations sur la disponibilité des données, voir le *Guide de l'utilisateur - Disques miroirs*.

4.5.3 Protection des bases de données avec RDDF7

Le logiciel RDDF7 (Fonction de duplication de base de données à distance) permet à un ordinateur de secours de conserver un double des bases de données ORACLE sur GCOS 7. RDDF7 peut être utilisé avec TDS, IOF ou des transactions de traitement par lots.

Pendant que l'application TDS s'exécute et que la base de données est mise à jour, RDDF7 transfère les images du journal Après sur le système de secours. Les images Après sont appliquées au double de la base de données. La duplication des mises à jour est différée sur le site à distance. Le logiciel RDDF7 garantit la cohérence des données entre les deux sites.

Si la machine principale est défaillante, TDS peut être lancé sur le système de secours en utilisant le double de la base de données. Lorsque la machine principale est relancée, les images Après peuvent être transférées sur le système restauré.

Pour plus d'informations sur la disponibilité des bases de données, voir le *Guide de l'utilisateur - RDDF7/RTO*.

4.5.4 SECUR'ACCESS

SECUR'ACCESS est un logiciel de contrôle d'accès logique destiné à protéger une application transactionnelle. Pour ce faire, il identifie, authentifie, puis autorise l'utilisateur à se connecter ou à se reconnecter à l'application. Ces contrôles peuvent aussi être programmés pour surveiller les accès utilisateur tout au long d'une session de travail.

Pour plus d'informations concernant l'utilisation de SECUR'ACCESS avec TDS, consulter les documents suivants :

- *SECUR'ACCESS - Guide de l'administrateur de sécurité*
- *SECUR'ACCESS - Guide de l'administrateur délégué*
- *SECUR'ACCESS - Guide de l'utilisateur*
- *SECUR'ACCESS - Manuel de programmation et de mise en oeuvre*

TDS - Concepts

Glossaire

Les termes qui apparaissent en gras dans une définition sont expliqués dans le glossaire.

administrateur

Voir administrateur TDS.

administrateur TDS

Personne responsable des fonctions administratives dans TDS, notamment de la configuration, de l'optimisation et de la mise en oeuvre d'une application TDS.

affectation d'un process

Attribution d'un process ou d'une tâche à une transaction.

bibliothèque

Fichier pouvant contenir des objets de différents types appelés **unités**. Une bibliothèque se divise fonctionnellement en sous-fichiers.

bibliothèque de modules partageables (SM)

Bibliothèque dans laquelle sont stockées les TPR compilées et reliées.

boîte aux lettres (Mailbox)

Adresse externe composée d'une chaîne de 8 caractères alphanumériques qui identifie un utilisateur, un correspondant, un terminal ou une imprimante.

La boîte aux lettres maître permet d'identifier l'opérateur maître.

boîte aux lettres maître (Master Mailbox)

Boîte aux lettres dont le nom (chaîne de 8 caractères alphanumériques) est catalogué en tant qu'application dans le catalogue de site avec et doté au minimum du code d'autorisation 0. La boîte aux lettres maîtresse permet d'identifier l'opérateur maître.

codes d'autorisation

Liste de codes qui autorisent ou empêchent les utilisateurs d'exécuter une transaction donnée *au sein de TDS*. Comme pour d'autres applications GCOS 7, tous les utilisateurs appartiennent à un ou plusieurs projets déclarés dans le catalogue de site (SITE.CATALOG). Les codes d'autorisation sont également déclarés dans le catalogue de site et affectés aux projets. Ainsi, par l'intermédiaire des codes d'autorisation associés aux projets, les utilisateurs peuvent accéder aux transactions.

commandes maître

Commandes réservées à l'opérateur maître, destinées à gérer TDS.

consolidation

Opération effectuée régulièrement par TDS à la fin normale d'un traitement. Elle intervient au niveau du point de reprise, c'est-à-dire lorsque :

- TDS sauvegarde l'ensemble du traitement de la transaction jusqu'à ce point : le traitement devient définitif et ne peut plus se perdre ni s'annuler,
- la transaction peut être relancée après un incident logiciel ou matériel,
- toutes les ressources affectées à la transaction sont libérées,
- les conflits d'accès aux ressources peuvent être résolus.

conversation

Période de temps séparant deux **échanges** d'une même **transaction**. C'est-à-dire temps compris entre la réception d'un message de sortie par l'utilisateur et l'émission du message d'entrée suivant introduit par cet utilisateur.

Le terme "conversation" a un sens différent dans un contexte XCP2.

correspondant

Application, utilisateur connecté à un terminal ou correspondant "virtuel" en communication avec une application TDS. Un correspondant utilise un protocole pour dialoguer avec une application TDS. Il peut s'agir du protocole TM (PT, interface traitement par lots), XCP1, XCP2 ou d'une session virtuelle.

correspondant virtuel

Correspondant "dummy". Il s'agit d'un correspondant qui n'est pas réel, et vu uniquement par TDS.

Il autorise le codage du traitement parallèle mis en oeuvre par le mécanisme de génération dynamique.

CRNETGEN

Utilitaire de génération de réseau et de définition de correspondants.

droits d'accès

Liste de codes déclarés pour chaque projet GCOS 7 qui autorisent (ou empêchent) les utilisateurs d'accéder aux disques, applications, environnements ou stations. Ces droits d'accès sont déclarés au *niveau du site* et ne doivent pas être confondus avec les codes d'autorisation qui concernent l'accès à une transaction TDS.

échange

Séquence comprenant le message d'entrée, le traitement par l'application et le message de sortie. Une routine TPR comprend au plus un échange, mais un échange peut être réparti sur un nombre quelconque de TPR. Une transaction peut comporter plusieurs échanges.

fenêtrage

Fonction permettant de recouvrir ou de superposer une *grille* ou fenêtre sur une autre.

fichier contrôlé par TDS

Fichier séquentiel indexé ou relatif UFAS ou UFAS-EXTENDED ou base de données IDS/II, contrôlés par TDS pendant la génération. Ce type de fichier permet le contrôle des accès concurrents.

fichier maître

Fichier contenant les données utilisateur de l'application. Il peut être considéré comme appartenant à la base de données. Un fichier maître est fréquemment utilisé et constamment mis à jour.

fichier non contrôlé

Fichier UFAS-EXTENDED (ou UFAS) qui n'est pas soumis au contrôle d'accès fourni par TDS.

fichiers en ligne

Fichiers utilisés pour le traitement des transactions. Ils doivent être disponibles (ou en ligne) lors de l'exécution.

fichiers hors ligne

Fichiers utilisés pour générer, compiler et relier TDS. Ils ne sont pas requis à l'exécution.

GAC-EXTENDED

Fonction de partage de fichier permettant des opérations de lecture et d'écriture sur le même fichier, et cela par différents utilisateurs concurrents dans divers environnements d'exploitation.

GCL (GCOS Command Language)

Langage de commande de GCOS.

génération dynamique

Lancement d'une transaction à partir d'une autre.

grille

Grille d'écran comportant des espaces (zones) à remplir par l'utilisateur.

GTWriter

Utilitaire GCOS 7 permettant de mettre en attente et d'imprimer des informations en mode asynchrone sur le terminal imprimante spécifié.

haute disponibilité

Produit logiciel GCOS 7 destiné à augmenter la disponibilité des applications. Egalement connu sous le nom de HA.

id-message

Nom utilisé pour identifier une transaction. Lorsque l'utilisateur introduit ce MESSAGE ou le sélectionne à partir d'un menu, la transaction correspondante est exécutée.

IDS/II

Base de données de réseau hiérarchique entièrement intégrée à TDS, fondée sur les fichiers UFAS.

image Après

Double de la mise à jour d'un fichier, enregistré dans le journal Après, destiné à reconstituer des fichiers en cas de **restauration actualisée (rollforward)**.

image Avant

Double du fichier contenant les données dans l'état précédant une mise à jour. Les images Avant sont enregistrées séquentiellement dans le journal Avant et sont utilisées en cas de restauration non actualisée (rollback).

interblocage

Conflit entre une ou plusieurs unités de consolidation requérant les ressources du système.

interface traitement par lots

Interface permettant d'établir une session et un dialogue entre un programme en traitement par lots et une application TDS.

Elle peut être utilisée comme fonction de mise au point de programme. Pour ce faire, un programme en traitement par lots simule un terminal, permettant au programme de se connecter en tant que terminal pour émettre et recevoir des messages.

IOF (Interactive Operation Facility)

Moniteur interactif IOF. Utilitaire GCOS 7 qui permet aux utilisateurs d'accéder aux informations (fichiers, bases de données, applications, programmes ou fonctions) et de soumettre des demandes.

JCL (Job Control Language)

Langage de gestion des travaux.

journal Après

Fonction GCOS 7 de protection contre les défaillances matérielles. Ce journal sert à enregistrer les mises à jour (images Après) à appliquer à un fichier ou une base de données.

journal Avant

Fonction GCOS 7 de protection contre les défaillances logicielles. Lorsqu'un article est traité, un double de l'article original (image Avant) est enregistré dans ce journal.

journal utilisateur

Journal privé dans lequel un utilisateur (ou à un programmeur) peut enregistrer des messages, des statistiques ou toute autre information utilisateur.

journalisation

Sélection de l'un des deux journaux de GCOS 7 (Avant et Après), ou des deux, pour protéger les fichiers.

lecture déstructurée ("sale")

Accès à un fichier en lecture pendant sa mise à jour. Les informations provenant d'une lecture déstructurée sont fiables ou non selon que la lecture a été effectuée avant ou après le point de consolidation de la TPR mettant à jour le fichier.

lecture exclusive

Type d'accès à un fichier dans lequel l'article est verrouillé pendant la lecture. Une seule routine TPR peut accéder à l'article à la fois : l'accès au fichier est "exclusif".

lecture partagée

Type d'accès à un fichier dans lequel l'article peut être lu par plusieurs TPR. Une lecture partagée ne consiste qu'à lire le fichier. Si la TPR doit écrire dans le fichier, elle doit attendre l'obtention d'une **lecture exclusive**.

lecture statistique

Lecture d'un fichier auquel les transactions peuvent accéder (en lecture et non en écriture) pendant sa mise à jour par d'autres consolidations.

mises à jour différées

Méthode de mise à jour par laquelle la modification du fichier n'est effectuée qu'à l'achèvement de l'**unité de consolidation**. Si une transaction est arrêtée prématurément avant la fin de l'unité de consolidation, la mise à jour n'est pas effectuée. Ce principe de mise à jour est mis en oeuvre par l'utilitaire **ROLLFORWARD**. Ce mécanisme améliore les performances mais nécessite des ressources en mémoire (tampons UFAS).

module édité

Résultat de l'édition de liens d'une routine TPR en COBOL. Les modules édités sont regroupés en modules partageables par l'éditeur de liens statique et stockés dans une bibliothèque de modules partageables (SM).

moniteur TDS

Logiciel qui gère l'exécution des programmes d'applications au cours du traitement de la transaction. Egalement connu sous le nom d'exécutif TDS.

MTGEN

Utilitaire JCL servant à gérer l'application TDS sous V3/V5. Sous V6, l'utilitaire correspondant est TP7GEN.

MTPREP

Utilitaire JCL chargé, sous V3/V5, de préparer l'application TDS en réservant l'espace pour les fichiers TDS. Sous V6, l'utilitaire correspondant est TP7PREP.

NETGEN

Voir CRNETGEN.

non-concurrence

Impossibilité d'activer simultanément des tâches appartenant à des transactions différentes. La déclaration de non-concurrence d'une transaction avec d'autres transactions (ou avec elle-même) s'effectue à la génération de TDS.

opérateur de terminal

Voir utilisateur final.

opérateur maître

Correspondant chargé de gérer la session TDS. Celui-ci doit avoir établi le dialogue de lancement avec TDS pour que les autres utilisateurs puissent se connecter.

ORACLE

Système de gestion de base de données relationnelles. ORACLE peut être utilisé avec TDS, mais il est géré par un logiciel indépendant.

pool

Ensemble de sessions parallèles ayant des caractéristiques communes. Elles peuvent avoir le même niveau de synchronisation ou les mêmes options de sécurité. Un pool doit contenir au moins une session. Un nombre maximum de sessions est défini pour chaque pool.

restauration actualisée (rollforward)

Méthode de reconstitution des fichiers en cas d'arrêt prématuré de TDS ou de crash système. Les fichiers sont ramenés à leur état le plus récent en appliquant aux articles les modifications sauvegardées en tant qu'images. Après.

restauration non actualisée (rollback)

Méthode de reconstitution d'un fichier à un état antérieur en ne tenant pas compte des interventions survenues depuis le dernier point de consolidation. L'image Avant remplace alors l'article mis à jour.

session

Connexion logicielle qui permet une suite de transmissions entre deux points, appelés **boîtes aux lettres**.

simultanéité

Niveau de fonctionnement multitâche autorisé dans une application TDS. Il s'agit du nombre maximum de tâches pouvant s'exécuter à un moment donné. La simultanéité est spécifiée à la génération de TDS.

STDS

Programme source TDS utilisé pour générer l'application TDS.

ystème couplé

Deux DPS 7000 dotés du produit systèmes couplés et partageant un ou plusieurs volumes disques.

TDSGEN

Programme de génération de TDS utilisé pour définir l'environnement dans lequel doivent s'exécuter les applications de traitement transactionnel.

TM (Terminal manager)

Protocole d'échange de données entre une application TDS et un terminal.

TP7GEN

Utilitaire JCL destiné à générer l'application TDS sous V6.

TP7PREP

Utilitaire JCL destiné à préparer l'application TDS en réservant de l'espace aux fichiers TDS (sous V6).

TPR (Transaction Processing Routine)

Routine de traitement de transaction. Unité de programme écrite en COBOL, en langage C ou en GPL par le programmeur TDS, et conçue pour s'exécuter dans l'environnement TDS. Une transaction est constituée d'une ou plusieurs TPR. Celles-ci sont stockées dans des modules partageables.

transaction

Unité de traitement définie par le programmeur TDS. Chaque transaction est composée d'une ou plusieurs routines de traitement de transaction (TPR). La mise à jour d'un compte client après paiement par carte de crédit est un exemple de transaction. Celle-ci comprend l'accès au fichier client, le dialogue avec l'opérateur et la modification du fichier.

transactions spéciales

Transactions écrites par le programmeur et appelées par TDS, lors du lancement ou de l'arrêt de TDS, lors de la connexion ou la déconnexion d'un utilisateur, de la déconnexion de l'utilisateur par TDS ou de sa relance, ou de lors de l'émission d'une interruption (BREAK).

unité

Sous-fichier de **bibliothèque**.

unité de consolidation

Partie du traitement écoulee depuis la dernière consolidation (ou début de la transaction s'il s'agit de la première unité de consolidation) jusqu'à la consolidation courante (ou fin de la transaction s'il s'agit de la dernière unité de consolidation). Egalement appelée UC.

utilisateur

Voir utilisateur final.

utilisateur final

Utilisateur dialoguant avec une application TDS via un terminal. Dans ce manuel, un utilisateur final est toute personne se connectant à TDS pour exécuter des transactions. Un utilisateur final n'est pas concerné par la mise au point ou l'administration d'une application TDS.

XCP1 (Extended Cooperative Protocol Level 1)

Protocole transactionnel coopératif étendu niveau 1. Protocole de communications entre applications standard.

XCP2 (Extended Cooperative Protocol Level 2)

Protocole transactionnel coopératif étendu niveau 2). Protocole de communications entre applications standard.

TDS - Concepts

F		Après	2-12
		Avant	2-12
fenêtres	3-14	utilisateur	2-12
fermeture de fichiers	3-22	journalisation	2-10
fichiers		journalisation TDS	2-10
accès	2-16		
contrôlés par TDS	2-16		
d'application	2-18		
différents types	2-16		
en ligne	2-18		
hors ligne	2-18		
non contrôlés par TDS	2-18		
TDS	2-16		
fonctions d'accès			
voir génération dynamique:	4-10		
FOR DEBUG	2-2		
FOR INQUIRY	2-2		
G		L	
GAC-EXTENDED	2-16	lancement	
génération		session utilisateur final	3-23
dynamique	4-10	TDS	3-18
gestion des états	3-15	transactions	3-23, 4-10
GPL	1-5	langage C	1-5
grilles	3-12	langages voir programmation	1-5
GTWriter	3-15	libération	
		process	4-6
		ressources	4-6
		ligne d'état	3-12
		limites	
		TPR simultanées	4-2
		transactions	4-2
		LINKER	3-8
		M	
		manipulation	
		données	2-7
		ressources	2-7
		messages externes	3-12
		méthode d'accès des	
		communications TDS	voir : TCAM 4-11
		mise au point des TPR	3-10
		mises à jour différées	2-13
		mode	
		actif	4-9
		fenêtrage	3-12
		grille	3-12
		normal	3-12
		passif	4-9
		module chargeable (LM)	3-15
		modules partageables voir SM	3-8
		moniteur TDS	1-4
		N	
		nombre	
		TPR	2-5
		transactions	2-2
		notions de TDS	2-1
		O	
		opérateur maître	1-9, 3-18
J			
journal			

Index

optimisation de TDS	4-1	utilisateur final	3-18
ORACLE avec TDS	4-20	simulation d'une application TDS voir interface	
organisation des données	4-19	traitement par lots de TDS	3-10
		simultanéité	
		niveau	4-2
P		SM	3-8
		sous-systèmes transactionnels voir TDS	1-5
point de consolidation	2-7	statistiques	2-12
point de consolidation implicite	2-7	en temps réel	3-10
procédures IQS	1-5	enregistrement	2-12
process	1-5, 4-6	SYSMANT	3-15
nombre	1-5		
produits de sécurité	4-21		
programmation	1-5	T	
programmation voir aussi transactions;	1-5	taux de traitement	4-2
programmeur TDS	1-9	TCAM	4-11
protection		TDS	
bases de données	4-22	assemblage	3-15
données	2-10	associé à d'autres produits	4-8
protocoles	4-11	avantages	1-5
Terminal Manager (TM)	4-11	documentation	1-10
XCP1	4-13	types d'utilisateurs	1-9
XCP2	4-15	TDS-HA	4-21
PT (Pass Thru) voir dispositif		temps de réponse	1-3
de déport TDS:	4-11	temps mort	2-5
		termes	2-1
		Terminal Adapter	3-12
R		Terminal Manager (TM)	4-11
		tests des TPR	3-10
RDDF7	4-22	TM (Terminal Manager)	4-11
reconstitution	2-10, 2-13	TPR	2-5
mécanismes	2-13	compilation	3-8
réduction des impacts d'incidents	4-21	écriture	3-7
relance de TDS	3-22	emplacement dans les modules	
Remote Database Duplication		partageables	3-8
Facility	voir :RDDF7	mise au point	3-10
	4-22	nombre limite	2-5
ressources	2-7	sauvegarde	3-8
restauration		test	3-10
actualisée (rollforward)	2-15	traitement	1-3
non actualisée (rollback)	2-14	données	4-2
rôle		en temps réel	1-1
opérateur maître	3-18	par lots	1-3
utilisateur final	3-23	transactions	1-1, 2-2
routine de traitement de		composants	2-3
transaction	voir : TPR2-5	différents types	2-2
		nombre limite	2-2
		non concurrentes	4-4
		simultanées	4-2
		spéciales	3-10
		Tx voir transaction:	2-2
S			
SECUR'ACCESS	4-23	U	
sécurité	1-5, 4-21	UC voir unité de consolidation	2-7
services multimédia	3-14	unité de consolidation	2-7
session			
arrêt	3-24		
communications	3-18, 4-11		
GCOS 7	3-18		
TDS	3-18		

TDS - Concepts

unité de traitement	
voir : unité de consolidation	2-7
UNIX	4-17
UPDATE	2-2
utilisateur	
final	1-9, 3-23
TDS	1-9
utilitaire de reconstitution	
du journal Après	2-13
utilitaires	
DUMPJRNL	2-13
journal	2-13
LINKER	3-8
ROLLFWD	2-13
SYSMANT	3-15

X

XCP1	4-13
XCP2	4-15

Z

zones de stockage	3-11
-------------------	------