



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

Q.787

(09/97)

SÉRIE Q: COMMUTATION ET SIGNALISATION

Spécifications du système de signalisation n° 7 –
Spécification des tests

**Spécification des tests relatifs au gestionnaire
de transaction**

Recommandation UIT-T Q.787

(Antérieurement Recommandation du CCITT)

RECOMMANDATIONS UIT-T DE LA SÉRIE Q

COMMUTATION ET SIGNALISATION

SIGNALISATION DANS LE SERVICE MANUEL INTERNATIONAL	Q.1–Q.3
EXPLOITATION INTERNATIONALE AUTOMATIQUE ET SEMI-AUTOMATIQUE	Q.4–Q.59
FONCTIONS ET FLUX D'INFORMATION DES SERVICES DU RNIS	Q.60–Q.99
CLAUSES APPLICABLES AUX SYSTÈMES NORMALISÉS DE L'UIT-T	Q.100–Q.119
SPÉCIFICATIONS DES SYSTÈMES DE SIGNALISATION N° 4 ET N° 5	Q.120–Q.249
SPÉCIFICATIONS DU SYSTÈME DE SIGNALISATION N° 6	Q.250–Q.309
SPÉCIFICATIONS DU SYSTÈME DE SIGNALISATION R1	Q.310–Q.399
SPÉCIFICATIONS DU SYSTÈME DE SIGNALISATION R2	Q.400–Q.499
COMMULATEURS NUMÉRIQUES	Q.500–Q.599
INTERFONCTIONNEMENT DES SYSTÈMES DE SIGNALISATION	Q.600–Q.699
SPÉCIFICATIONS DU SYSTÈME DE SIGNALISATION N° 7	Q.700–Q.849
Généralités	Q.700
Sous-système transport de messages	Q.701–Q.709
Sous-système commande des connexions sémaphores	Q.711–Q.719
Sous-système utilisateur de téléphonie	Q.720–Q.729
Services complémentaires du RNIS	Q.730–Q.739
Sous-système utilisateur de données	Q.740–Q.749
Gestion du système de signalisation n° 7	Q.750–Q.759
Sous-système utilisateur du RNIS	Q.760–Q.769
Sous-système application de gestion des transactions	Q.770–Q.779
Spécification des tests	Q.780–Q.799
Interface Q3	Q.800–Q.849
SYSTÈME DE SIGNALISATION D'ABONNÉ NUMÉRIQUE N° 1	Q.850–Q.999
RÉSEAUX MOBILES TERRESTRES PUBLICS	Q.1000–Q.1099
INTERFONCTIONNEMENT AVEC LES SYSTÈMES MOBILES À SATELLITES	Q.1100–Q.1199
RÉSEAU INTELLIGENT	Q.1200–Q.1999
RNIS À LARGE BANDE	Q.2000–Q.2999

Pour plus de détails, voir la Liste des Recommandations de l'UIT-T.

RECOMMANDATION UIT-T Q.787

SPÉCIFICATION DES TESTS RELATIFS AU GESTIONNAIRE DE TRANSACTION

Résumé

La présente Recommandation révisée Q.787 contient les descriptions de tests des capacités de transaction (TC, *transaction capabilities*) du système de signalisation n° 7. La présente Recommandation révisée couvre aujourd'hui les descriptions de test pour la Partie Dialogue du *Livre Blanc* (1993) Recommandations Q.771 à Q.774 (Sous-système application du gestionnaire de transaction).

Source

La Recommandation UIT-T Q.787, révisée par la Commission d'études 11 de l'UIT-T (1997-2000), a été approuvée le 12 septembre 1997 selon la procédure définie dans la Résolution n° 1 de la CMNT.

AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

La Conférence mondiale de normalisation des télécommunications (CMNT), qui se réunit tous les quatre ans, détermine les thèmes d'études à traiter par les Commissions d'études de l'UIT-T lesquelles élaborent en retour des Recommandations sur ces thèmes.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution n° 1 de la CMNT.

Dans certains secteurs de la technologie de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI.

NOTE

Dans la présente Recommandation, l'expression "Administration" est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou la mise en œuvre de la présente Recommandation puisse donner lieu à l'utilisation d'un droit de propriété intellectuelle. L'UIT ne prend pas position en ce qui concerne l'existence, la validité ou l'applicabilité des droits de propriété intellectuelle, qu'ils soient revendiqués par un Membre de l'UIT ou par une tierce partie étrangère à la procédure d'élaboration des Recommandations.

A la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux responsables de la mise en œuvre de consulter la base de données des brevets du TSB.

© UIT 1998

Droits de reproduction réservés. Aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'UIT.

TABLE DES MATIÈRES

	Page
1	Introduction 1
2	Objectifs de la spécification des tests..... 1
3	Domaine d'application..... 1
4	Principes généraux de tests..... 2
5	Contexte d'exécution des tests..... 2
5.1	Relation sémaphore 2
5.2	Configuration..... 2
6	Trafic de base 3
7	Liste de tests 3
7.1	Spécification de tests de la sous-couche transaction TC 3
7.1.1	Directives relatives à l'exécution des tests de la couche Transaction 3
7.1.2	Tests de la sous-couche transaction 3
7.2	Spécification de tests de la sous-couche composant TC (CSL) 114
7.2.1	Directives relatives à l'exécution des tests de la sous-couche composant.... 114
7.2.2	Liste de tests de la sous-couche composant 115

Recommandation Q.787

SPÉCIFICATION DES TESTS RELATIFS AU GESTIONNAIRE DE TRANSACTION

(révisée en 1997)

1 Introduction

La présente Recommandation contient un ensemble détaillé de tests des relatifs aux gestionnaires de transactions (TC, *transaction capabilities*) du système de signalisation n° 7. Ces tests ont pour but de valider le protocole spécifié dans les Recommandations Q.771 à Q.774. La présente Recommandation est conforme au *Livre blanc* (1993) qui décrit les règles de base d'une spécification de tests, telles qu'elles sont énoncées dans la Recommandation Q.780.

2 Objectifs de la spécification des tests

La spécification des tests permet de s'assurer avec un certain niveau de confiance:

Validation – Qu'un mode de réalisation donné est conforme aux Recommandations Q.771 à Q.774 du *Livre blanc* (1993) pour les gestionnaires de transactions du SS n° 7.

Compatibilité – Que l'interfonctionnement de deux modes de réalisation des gestionnaires de transactions du SS n° 7 est possible.

Les critères suivants ont été utilisés pour l'élaboration de la présente spécification de tests:

- 1) la spécification des tests ne couvre pas de manière exhaustive tous les aspects des gestionnaires de transactions du SS n° 7;
- 2) tous les tests ont un caractère pratique et peuvent être exécutés à l'aide des techniques existantes;
- 3) la liste des tests porte essentiellement sur des procédures de signalisation normales. Les tests relatifs à des procédures de signalisation anormales ne sont identifiés que lorsque cela est considéré comme particulièrement utile;
- 4) la liste des tests ne comporte pas de tests particuliers à une application donnée. De tels tests doivent être décrits dans une documentation de test relative à cette application et sortent du cadre de la présente spécification.

3 Domaine d'application

Les descriptions de test se divisent en deux sous-paragraphe: 7.1 spécification de test de la sous-couche transaction du gestionnaire de transaction (TSL, *transaction sublayer*) et 7.2 spécification de test de la sous-couche composant du gestionnaire de transaction (CSL, *component sublayer*). La plupart des fonctions TSL et CSL dépendent les unes des autres et devront être exécutées conjointement. La distinction entre fonctions TSL et fonctions CSL a pour seul but de clarifier les tests et d'en faciliter la compréhension; elle n'implique aucune mise en œuvre particulière.

La présente spécification de tests est destinée à vérifier les fonctionnalités du TCAP (sous-système application pour la gestion des transactions) en testant les messages du TCAP et leur contenu. Elle ne tient pas compte des aspects de fonctionnement tels que la limitation du nombre d'identificateurs (ID) de transaction.

Certains tests décrits dans la présente Recommandation nécessitent la création de primitives; lors de l'exécution de ces tests, l'utilisateur du sous-système TCAP devra donc choisir, dans le cadre du système utilisé, des actions normales appropriées qui auront pour effet de créer les primitives indiquées.

Le test des primitives sort du cadre de la présente Recommandation. Les messages et les primitives sont inclus dans les diagrammes de séquence de messages attendue, comme indiqué ci-dessous, mais les primitives ne sont représentées que pour faciliter la compréhension des tests.

PRIMITIVE: =====>

MESSAGE: —————>

La description de test constitue un guide pour l'interprétation et la mise en œuvre correctes du test, mais elle n'impose aucune contrainte concernant la réalisation de ce test. En particulier, il n'est fait référence à la structure interne de l'implémentation sous test (IUT, *implementation under test*), telle que la confirmation des états internes des automates à états des gestionnaires de transactions, qu'à titre de clarification et la réalisation pratique de cette structure peut dépendre de l'application ou varier d'un test à l'autre. Pour un fonctionnement correct, une réponse positive doit être donnée à toutes les questions et vérifications contenues dans la description de test.

Il est fait mention, dans toute la spécification de tests, d'"automates à états". Ce modèle conceptuel de la spécification est utilisé dans la Recommandation Q.774 pour faciliter la compréhension des tests. Il n'implique aucun mode de mise en œuvre particulier même lorsque, dans la description du test, il est demandé que l'état soit confirmé à la fin de certains tests.

Les "Directives relatives à l'exécution des tests de la sous-couche composant" énoncées en 7.1.1 et 7.2.1 indiquent des méthodes possibles permettant de s'assurer que le logiciel est revenu à l'état requis.

La spécification de tests est indépendante de toute application ou mise en œuvre particulière.

4 Principes généraux de tests

Les tests se présentent sous la forme de tests de "Validation" ou de "Validation et Compatibilité". Chaque description de test indique, à la rubrique "Type de test", s'il s'agit d'un test "VAT" (Validation) ou "VAT et CPT" (Validation et Compatibilité).

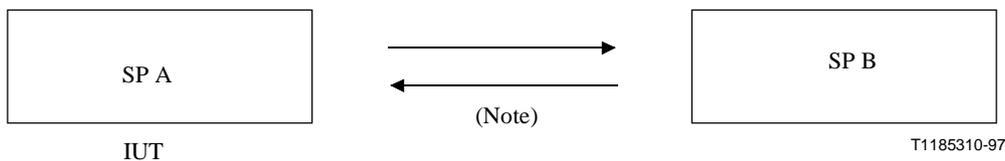
5 Contexte d'exécution des tests

5.1 Relation sémaphore

Une relation sémaphore stable est nécessaire entre les points sémaphores "SP A" et "SP B" pour tester efficacement le TCAP. Une couche Service réseau testée, par exemple une relation sémaphore de MTP ou SCCP, doit être utilisée pour les tests de compatibilité.

5.2 Configuration

Une seule configuration, représentée dans la Figure 1, est nécessaire pour effectuer les tests indiqués dans la liste de tests proposée:



NOTE – Les flèches indiquent une relation sémaphore.

Figure 1/Q.787 – Configuration: 1

6 Trafic de base

Les tests décrits ne tiennent aucunement compte du niveau du trafic de base.

7 Liste de tests

Les catégories de la liste de tests sont indiquées dans les sous-paragraphe ci-dessous.

7.1 Spécification de tests de la sous-couche transaction TC

7.1.1 Directives relatives à l'exécution des tests de la couche Transaction

La séquence de messages attendue, une description du test et une table de vérification des éléments d'information (IE, *information elements*) sont présentés pour chaque test.

Dans la séquence de messages attendue, les primitives sont indiquées au point SP A [côté implémentation sous test (IUT)] seulement.

La table de vérification a pour but d'indiquer à la fois le contenu du message de lancement et les résultats attendus pour effectuer les vérifications spécifiées dans les descriptions de test. La table de vérification des IE contenus dans les messages ne comporte pas d'informations sur le contenu des IE partie composant (*component portion*) ou abandon par l'utilisateur (*user abort*) qui dépend de l'application. Dans les tables de vérification, les messages émanant de l'IUT sont décrits sous la forme courte pour toute longueur d'IE, sauf en 1.1.3.1.1 qui concerne les tests des variations de longueur. Cependant, différentes formes répondant aux conditions du 3.3/Q.773, peuvent être utilisées, quel que soit le test.

Il est suggéré d'utiliser la procédure suivante pour vérifier les résultats antérieurs et postérieurs aux tests (par exemple, automates à l'état de repos):

- envoi à l'IUT d'un message Continue (Continuation) avec Identificateur (ID) de transaction de destination identique (transaction qui devrait être inactive) et réception prévisible d'un message Abort (ABANDON) avec une valeur de cause ID de transaction non reconnu. Si un autre message est reçu en réponse, ceci signifie alors que la transaction n'est pas à l'état de repos.

NOTE – Les détails de ces tests de confirmation dépendent du mode de réalisation.

7.1.2 Tests de la sous-couche transaction

NDA aucun détail disponible (*no details available*)

FFS à étudier (*for further study*)

* validation et compatibilité

Tous les autres tests sont des tests de validation seulement.

1 *Sous-couche Transaction*

1.1 Fonction valide

1.1.1 Dialogue non structuré

* 1.1.1.1 Côté testé: émission

* 1.1.1.2 Côté testé: réception

1.1.2 Dialogue structuré

1.1.2.1 Libération avant message ultérieur

1.1.2.1.1 Libération valide émanant du côté initialisation

* 1) Terminaison prédéterminée

* 2) Abandon par l'Utilisateur-TR

1.1.2.1.2 Libération valide émanant du côté réponse

1.1.2.1.2.1 Emission IUT

* 1) Terminaison de base

* 2) Terminaison prédéterminée

* 3) Abandon par l'Utilisateur-TR

1.1.2.1.2.2 Réception IUT

* 1) Abandon par l'Utilisateur-TR

2) Abandon par la sous-couche transaction

3) Terminaison de base

1.1.2.2 Libération après un message Continue

1.1.2.2.1 Libération valide émanant du côté initialisation

1.1.2.2.1.1 Emission IUT

* 1) Terminaison de base

* 2) Terminaison prédéterminée

* 3) Abandon par l'Utilisateur-TR

1.1.2.2.1.2 Réception IUT

* 1) Terminaison de base

2) Abandon par la sous-couche transaction

3) Abandon par l'Utilisateur-TR

1.1.2.2.2 Libération valide émanant du côté réponse

1.1.2.2.2.1 Emission IUT

* 1) Terminaison de base

* 2) Terminaison prédéterminée

* 3) Abandon par l'Utilisateur-TR

1.1.2.2.2.2 Réception IUT

* 1) Terminaison de base

2) Abandon par la sous-couche transaction

3) Abandon par l'Utilisateur-TR

1.1.2.3 Libération après un message Continue (partie composant non présente)

1.1.2.3.1 Terminaison de base – Emission IUT

1.1.2.3.2 Terminaison de base – Réception IUT

1.1.2.4 Echange de messages après l'établissement de la transaction

1.1.2.4.1 Lancement IUT

- 1.1.2.4.2 Réception IUT
- 1.1.2.5 Adressage TC
 - 1.1.2.5.1 Changement d'adresse de Registre
- 1.1.3 Variations de codage et de valeur
 - 1.1.3.1 Variations de codage
 - 1.1.3.1.1 Variations de longueur
 - 1.1.3.1.1.1 Forme courte définie
 - 1) Longueur de partie Composant sous forme courte définie intégrée dans la forme courte
 - 2) Longueur de partie Composant sous forme courte définie intégrée dans la forme longue
 - 1.1.3.1.1.2 Forme longue définie
 - 1) Longueur de partie Composant sous forme longue définie intégrée dans la forme longue
 - 1.1.3.1.1.3 Forme indéfinie
 - 1) Longueur de partie Composant sous forme indéfinie intégrée dans la forme indéfinie
 - 1.1.3.2 Variations de valeur
 - 1.1.3.2.1 ID de transaction
 - 1) Longueur d'un octet
 - 2) Longueur de quatre octets

*

*

1.2 Comportement syntaxiquement non valide

- 1.2.1 Valeurs non valides pour les éléments d'information
 - 1.2.1.1 Type de message Begin
 - 1) Longueur d'OTID (identificateur de transaction d'origine)= 0
 - 2) Longueur d'OTID > quatre octets
 - 1.2.1.2 Premier message Continue
 - 1) Longueur de DTID (identificateur de transaction de destination) = 0
 - 1.2.1.3 Message Continue ultérieur
 - 1) Longueur de partie Composant incorrecte
 - 1.2.1.4 Message End (fin)
 - 1) Longueur de DTID > quatre octets
 - 1.2.1.5 Message Abort (abandon)
 - 1) Valeur de cause: P-Abort (Abandon par le fournisseur) non valide
 - 2) Longueur de cause P-Abort incorrecte
- 1.2.2 Structure non valide
 - 1.2.2.1 Type de message Unidirectionnel
 - 1) Élément d'information inconnu présent
 - 1.2.2.2 Type de message Begin
 - 1) OTID absent
 - 2) Élément d'information inconnu présent
 - 1.2.2.3 Premier message Continue

- 1) OTID absent
- 2) DTID absent
- 3) OTID dupliqué
- 4) DTID dupliqué
- 5) Élément d'information inconnu présent
- 1.2.2.4 Message Continue ultérieur
 - 1) OTID absent
 - 2) Élément d'information inconnu présent
- 1.2.2.5 Message End
 - 1) DTID absent
- 1.2.2.6 Message Abort
 - 1) DTID absent
- 1.2.2.7 Message Unknown (inconnu)
 - 1) OTID non inclus
 - 2) OTID inclus et DTID non inclus
 - 3) OTID inclus et DTID inclus
- 1.2.3 Codage non valide (c'est-à-dire violation du BER de la Rec. X.209)
 - 1.2.3.1 Type de message Begin
 - 1) Etiquette non valide
 - 1.2.3.2 Type de message Continue
 - 1) Etiquette non valide
- 1.3 Messages inopportuns
 - 1.3.1 Type de message Continue
 - 1) Réception d'un message Continue à l'état de repos avec DTID non affecté
 - 1.3.2 Type de message End
 - 1) Réception d'un message End à l'état de repos
 - 1.3.3 Type de message Abort
 - 1) Réception d'un message Abort à l'état de repos
- 1.4 Codage de transactions multiples
 - 1.4.1 Codage de transaction valide
 - 1) Nouvelle demande de transaction pendant l'établissement de la transaction
 - 2) Nouvelle demande de transaction après l'établissement de la transaction
 - 1.4.2 Messages inopportuns
 - 1) Message avec DTID non affecté pendant l'établissement de la transaction
 - 2) Message avec DTID non affecté après l'établissement de la transaction

NUMÉRO DU TEST: 1.1.1.1		Feuille: 1 de 1
RÉFÉRENCE: 3.3.3.1.1/Q.774		
TITRE: Fonction valide; Dialogue non structuré		
SOUS-TITRE: Côté testé: émission		
OBJET: Vérifier que le point sémaphore A peut recevoir correctement un message Unidirectionnel		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (TSL) SP B (TSL)</p> <p><i>dem. TR-UNI</i></p> <p>=====></p> <p>UNIDIRECTIONAL —————></p>		
DESCRIPTION DU TEST		
1.	Envoi d'un message Unidirectionnel du point SP A au point SP B.	
2.	VÉRIFICATION A: LE MESSAGE UNIDIRECTIONAL A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
3.	VÉRIFICATION B: L'AUTOMATE TSL ASSOCIÉ À CETTE TRANSACTION A-T-IL ÉTÉ LAISSÉ À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
<p>UNIDIRECTIONAL</p> <p>Etiquette de type de message: 01100001</p> <p>Longueur de type de message: nombre correct d'octets</p> <p>Etiquette de partie composant: 01101100</p> <p>Longueur de partie composant: nombre correct d'octets</p>		

NUMÉRO DU TEST: 1.1.1.2		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.3.1.2/Q.774		
TITRE: Fonction valide; Dialogue non structuré		
SOUS-TITRE: Côté testé: réception		
OBJET: Vérifier que le point sémaphore A peut recevoir correctement un message Unidirectionnel		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (TSL) SP B (TSL)</p> <p style="text-align: center;">←—————</p> <p style="text-align: right;">UNIDIRECTIONAL</p> <p><i>ind. TR-UNI</i></p> <p>←=====</p>		
DESCRIPTION DU TEST		
1.	Envoi d'un message Unidirectionnel du point SP B au point SP A.	
2.	VÉRIFICATION A: LE MESSAGE UNIDIRECTIONAL A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
3.	VÉRIFICATION B: L'AUTOMATE TSL ASSOCIÉ À CETTE TRANSACTION A-T-IL ÉTÉ LAISSÉ À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
UNIDIRECTIONAL		
Etiquette de type de message: 01100001		
Longueur de type de message: nombre correct d'octets		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		

NUMÉRO DU TEST: 1.1.2.1.1 1)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération avant message ultérieur; Libération valide émanant du côté initialisation; Terminaison prédéterminée		
OBJET: Vérifier que le point sémaphore A peut envoyer correctement un message Begin puis terminer la transaction localement par la méthode "terminaison prédéterminée"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL)		SP B (TSL)
<i>dem. TR-BEGIN</i>		
=====>		
BEGIN	—————>	
<i>dem. TR-END</i>		
=====>		
<i>(Terminaison prédéterminée)</i>		
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B.	
2.	Avant la réception d'une réponse émanant du point SP B, faire en sorte que la primitive de demande TR-END (terminaison prédéterminée) soit transmise à la sous-couche transaction au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: VÉRIFIER QU'UN MESSAGE END N'A PAS ÉTÉ ENVOYÉ PAR LE POINT SP A.	
5.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN		
Etiquette de type de message: 01100010		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de départ: 01001000		
Longueur d'ID de transaction de départ: nombre correct d'octets		
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		

NUMÉRO DU TEST: 1.1.2.1.1 2)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.3.2.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération avant message ultérieur; Libération valide émanant du côté initialisation; Abandon par l'utilisateur-TR		
OBJET: Vérifier que le point sémaphore A peut émettre correctement un message Begin puis terminer la transaction localement par la méthode "abandon"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL)		SP B (TSL)
<i>dem. TR-BEGIN</i>		
=====>		
BEGIN	—————>	
<i>dem. TR-U-ABORT</i>		
=====>		
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B.	
2.	Avant la réception d'une réponse émanant du point SP B, faire en sorte qu'une primitive de demande TR-U-ABORT soit transmise à la sous-couche TSL au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: LA DEMANDE TR-U-ABORT ÉTAIT-ELLE PUREMENT LOCALE AU POINT SP A?	
5.	VÉRIFICATION C: VÉRIFIER QU'AUCUN MESSAGE "ABORT" N'A ÉTÉ ENVOYÉ PAR LE POINT SP A.	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN		
Etiquette de type de message: 01100010		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de départ: 01001000		
Longueur d'ID de transaction de départ: nombre correct d'octets		
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		

NUMÉRO DU TEST: 1.1.2.1.2.1 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération avant message ultérieur; Libération valide émanant du côté réponse; Emission IUT; Terminaison de base		
OBJET: Vérifier que le point sémaphore A peut recevoir correctement un message Begin puis terminer la transaction par la méthode "terminaison de base"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>ind. TR-BEGIN</i> <===== <i>dem. TR-END</i> =====> (Terminaison de base) END	←————— —————→	SP B (TSL) BEGIN
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP B au point SP A.	
2.	A la réception de l'indication BEGIN, faire en sorte qu'une primitive de demande TR-END (terminaison de base) soit transmise à la sous-couche transaction au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ CORRECTEMENT REÇU AU POINT SP A ET TRANSMIS À L'UTILISATEUR-TR?	
4.	VÉRIFICATION B: UN MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION C: LE DTID DANS LE MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
6.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.1.2.1 2)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération avant message ultérieur; Libération valide émanant du côté réponse; Emission IUT; Terminaison prédéterminée		
OBJET: Vérifier que le point sémaphore A peut recevoir un message Begin puis terminer la transaction localement par la méthode "terminaison prédéterminée"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL) SP B (TSL)</p> <p style="text-align: center;">←————— BEGIN</p> <p><i>ind. TR-BEGIN</i> ←</p> <p><i>dem. TR-END</i> →</p> <p>(Terminaison prédéterminée)</p>		
DESCRIPTION DU TEST		
<ol style="list-style-type: none"> 1. Envoi d'un message Begin du point SP B au point SP A. 2. A la réception de l'indication BEGIN, faire en sorte qu'une primitive de demande TR-END (terminaison prédéterminée) soit transmise à la sous-couche transaction au point SP A. 3. VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A? 4. VÉRIFICATION B: VÉRIFIER QU'UN MESSAGE END N'A PAS ÉTÉ ENVOYÉ PAR LE POINT SP A. 5. VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A? 		
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN Etiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets Etiquette d'ID de transaction de départ: 01001000 Longueur d'ID de transaction de départ: nombre correct d'octets Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets		

NUMÉRO DU TEST: 1.1.2.1.2.2 1)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.3.2.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération avant message ultérieur; Libération valide émanant du côté réponse; Réception IUT; Abandon par l'utilisateur-TR		
OBJET: Vérifier que le point sémaphore A peut terminer une transaction à la réception d'un message Abort (U).		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>ind. TR-U-ABORT</i></p> <p><=====</p>	<p>—————→</p> <p>←—————</p>	<p>SP B (TSL)</p> <p>ABORT (U)</p>
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B.	
2.	Faire en sorte que le point SP B envoie un message U-Abort au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: LE MESSAGE ABORT A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN		
Etiquette de type de message: 01100010		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de départ: 01001000		
Longueur d'ID de transaction de départ: nombre correct d'octets		
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
ABORT (U)		
Etiquette de type de message: 01100111		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de destination: 01001001		
Longueur d'ID de transaction de destination: nombre correct d'octets		
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) (valeur de l'OTID reçue dans le message BEGIN)		
Etiquette d'information abandon par l'utilisateur: 01101011		
Longueur d'information abandon par l'utilisateur: nombre correct d'octets		

NUMÉRO DU TEST: 1.1.2.1.2.2 2)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération avant message ultérieur; Libération valide émanant du côté réponse; Réception IUT; Abandon par la sous-couche transaction		
OBJET: Vérifier que le point sémaphore A peut terminer une transaction à la réception d'un message Abort (P)		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (TSL) SP B (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN —————→</p> <p>←————— ABORT (P) <i>ind. TR-P-ABORT</i> <=====</p>		
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B.	
2.	Faire en sorte que le point SP B envoie un message P-Abort au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: LE MESSAGE ABORT A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
<p>BEGIN</p> <p>Etiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette d'ID de transaction de départ: 01001000 Longueur d'ID de transaction de départ: nombre correct d'octets Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>ABORT (P)</p> <p>Etiquette de type de message: 01100111 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette d'ID de transaction de destination: 01001001 Longueur d'ID de transaction de destination: nombre correct d'octets Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) (valeur de l'OTID reçue dans le message BEGIN)</p> <p>Etiquette de cause P-Abort: 01001010 Longueur de cause P-Abort: un octet Valeur de cause P-Abort: entier (entre 0 et 4)</p>		

NUMÉRO DU TEST: 1.1.2.1.2.2.3)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération avant message ultérieur; Libération valide émanant du côté réponse; Réception IUT; Terminaison de base		
OBJET: Vérifier que le point sémaphore A peut terminer une transaction à la réception d'un message End		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL) SP B (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN —————→</p> <p>←————— END</p> <p><i>ind. TR-END</i> <=====</p>		
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B.	
2.	Faire en sorte que le point SP B envoie un message End au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: LE MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN		
Etiquette de type de message: 01100010		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de départ: 01001000		
Longueur d'ID de transaction de départ: nombre correct d'octets		
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
END		
Etiquette de type de message: 01100100		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de destination: 01001001		
Longueur d'ID de transaction de destination: nombre correct d'octets		
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) (valeur de l'OTID reçue dans le message BEGIN)		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		

NUMÉRO DU TEST: 1.1.2.2.1.1 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774, 3.3.3.2.2/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté initialisation; Emission IUT; Terminaison de base		
OBJET: Vérifier que le point sémaphore A peut terminer la transaction par la méthode "terminaison de base"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Le point SP B doit répondre par un message Continue à la réception du message Begin.		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>_____→</p> <p>←_____</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>dem. TR-END</i></p> <p>=====></p> <p><i>(Terminaison de base)</i></p> <p>END</p> <p>_____→</p>		<p>SP B (TSL)</p> <p>CONTINUE</p>
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B. Faire en sorte que le point SP B réponde par un message Continue.	
2.	A la réception de l'indication CONTINUE, faire en sorte qu'une primitive de demande TR-END (terminaison de base) soit transmise à la sous-couche transaction au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LA SOUS-COUCHE TRANSACTION AU POINT SP A?	
5.	VÉRIFICATION C: LE MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION D: LE DTID DANS LE MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE CONTINUE?	
7.	VÉRIFICATION E: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message CONTINUE)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.1.1 2)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774, 3.3.3.2.2/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté initialisation; Emission IUT; Terminaison prédéterminée		
OBJET: Vérifier que le point sémaphore A peut terminer la transaction par la méthode "terminaison prédéterminée"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Le point SP B doit répondre par un message Continue à la réception du message Begin		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>_____→</p> <p>←_____</p> <p><i>ind. TR-CONTINUE</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====></p> <p>(Terminaison prédéterminée)</p>		<p>SP B (TSL)</p> <p>CONTINUE</p>
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B. Faire en sorte que le point SP B réponde par un message Continue.	
2.	A la réception de l'indication CONTINUE, faire en sorte qu'une primitive de demande TR-END (terminaison prédéterminée) soit transmise à la sous-couche transaction au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION C: VÉRIFIER QUE LA PRIMITIVE DE DEMANDE TR-END ÉTAIT PUREMENT LOCALE ET QU'UN MESSAGE END N'A PAS ÉTÉ CRÉÉ ET ENVOYÉ PAR LE POINT SP A.	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.1.1 3)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774, 3.3.3.2.2/Q.774 et 3.3.3.2.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté initialisation; Emission IUT; Abandon par l'utilisateur-TR		
OBJET: Vérifier que le point sémaphore A peut terminer la transaction par la méthode "abandon"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Le point SP B doit répondre par un message Continue à la réception du message Begin		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p>←————</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>dem. TR-U-ABORT</i></p> <p>=====></p> <p>ABORT (U)</p> <p>—————→</p>		<p>SP B (TSL)</p> <p>CONTINUE</p>
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B. Faire en sorte que le point SP B réponde par un message Continue.	
2.	A la réception de l'indication CONTINUE, faire en sorte qu'une primitive de demande TR-U-ABORT soit transmise à la sous-couche transaction au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION C: LE MESSAGE ABORT A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION D: LE DTID DANS LE MESSAGE ABORT ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE CONTINUE?	
7.	VÉRIFICATION E: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

ABORT

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message CONTINUE)
Etiquette d'information abandon par l'utilisateur: 01101011
Longueur d'information abandon par l'utilisateur: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.1.2 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1.2/Q.774, 3.3.3.2.2/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté initialisation; Réception IUT; Terminaison de base		
OBJET: Vérifier que le point sémaphore A peut créer un message Continue puis terminer la transaction à la réception d'un message End		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====→</p> <p>CONTINUE</p> <p><i>ind. TR-END</i></p> <p>←=====</p>	<p>←-----</p> <p>-----→</p> <p>←-----</p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>END</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A.	
2.	Faire en sorte que le point SP A réponde par un message Continue.	
3.	Faire en sorte que le point SP B réponde par un message End.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION C: LE MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
7.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message CONTINUE)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.1.2 2)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1.2/Q.774 et 3.3.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Après un message Continue; Libération valide émanant du côté initialisation; Réception IUT; Abandon par la sous-couche transaction		
OBJET: Vérifier que le point sémaphore A peut créer un message Continue puis terminer la transaction à la réception d'un message Abort par la sous-couche transaction homologue		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====→</p> <p>CONTINUE</p> <p><i>ind. TR-P-ABORT</i></p> <p>←=====</p>	<p>←-----</p> <p>-----→</p> <p>←-----</p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>ABORT (P)</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A.	
2.	Faire en sorte que le point SP A réponde par un message Continue.	
3.	Faire en sorte que le point SP B réponde par un message Abort (P).	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION C: LE MESSAGE ABORT A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
7.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message CONTINUE)
Etiquette de cause P-Abort: 01001010
Longueur de cause P-Abort: un octet
Valeur de cause P-Abort: entier (0 à 4)

NUMÉRO DU TEST: 1.1.2.2.1.2 3)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1.2/Q.774 et 3.3.3.2.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté initialisation; Réception IUT; Abandon par l'utilisateur-TR		
OBJET: Vérifier que le point sémaphore A peut créer un message Continue puis terminer la transaction à la réception d'un message Abort par l'utilisateur-TR homologue		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====→</p> <p>CONTINUE</p> <p><i>ind. TR-U-ABORT</i></p> <p>←=====</p>	<p>←-----</p> <p>-----→</p> <p>←-----</p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>ABORT (U)</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A.	
2.	Faire en sorte que le point SP A réponde par un message Continue.	
3.	Faire en sorte que le point SP B réponde par un message Abort (U).	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION C: LE MESSAGE ABORT A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
7.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

ABORT (U)

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message CONTINUE)
Etiquette d'information abandon par l'utilisateur: 01101011
Longueur d'information abandon par l'utilisateur: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.2.1 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774, 3.3.3.2.2/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté réponse; Emission IUT; Terminaison de base		
OBJET: Vérifier que le point sémaphore A peut créer un message Continue puis terminer la transaction par la méthode "terminaison de base"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====→</p> <p>CONTINUE</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p><i>(Terminaison de base)</i></p> <p>END</p>	<p>←-----</p> <p>-----→</p> <p>-----→</p>	<p>SP B (TSL)</p> <p>BEGIN</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A.	
2.	Faire en sorte que le point SP A réponde par un message Continue.	
3.	Terminer la transaction par un message End (terminaison de base) émanant du point SP A.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LA SOUS-COUCHE TRANSACTION AU POINT SP A?	
6.	VÉRIFICATION C: UN MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
7.	VÉRIFICATION D: LE DTID DANS LES MESSAGES CONTINUE ET END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
8.	VÉRIFICATION E: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

END

Etiquette de type de message: 01100100
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message CONTINUE)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.2.1 2)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774, 3.3.3.2.2/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté réponse; Emission IUT; Terminaison prédéterminée		
OBJET: Vérifier que le point sémaphore A peut créer un message Continue puis terminer la transaction par la méthode "terminaison prédéterminée"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====→</p> <p>CONTINUE</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p>(Terminaison prédéterminée)</p>	<p>←-----</p> <p>BEGIN</p> <p>-----→</p>	<p>SP B (TSL)</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A.	
2.	Faire en sorte que le point SP A réponde par un message Continue.	
3.	Terminer la transaction par une primitive de demande TR-END (terminaison prédéterminée) émanant du point SP A.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LA SOUS-COUCHE TRANSACTION AU POINT SP A?	
6.	VÉRIFICATION C: VÉRIFIER QUE LA PRIMITIVE DE DEMANDE TR-END ÉTAIT PUREMENT LOCALE ET QU'UN MESSAGE END N'A PAS ÉTÉ CRÉÉ ET ENVOYÉ PAR LE POINT SP A.	
7.	VÉRIFICATION D: LE DTID DANS LE MESSAGE CONTINUE ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
8.	VÉRIFICATION E: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.1 3)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774, 3.3.3.2.2/Q.774 et 3.3.3.2.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté réponse; Emission IUT; Abandon par l'utilisateur-TR		
OBJET: Vérifier que le point sémaphore A peut créer un message Continue puis terminer la transaction par la méthode "abandon"		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====→</p> <p>CONTINUE</p> <p><i>dem. TR-U-ABORT</i></p> <p>=====→</p> <p>ABORT (U)</p>	<p>←-----</p> <p>BEGIN</p> <p>-----→</p> <p>-----→</p>	<p>SP B (TSL)</p>
DESCRIPTION DU TEST		
1.	Faire en sorte qu'un message Begin soit envoyé du point SP B au point SP A.	
2.	Faire en sorte que le point SP A réponde par un message Continue, puis abandonne la transaction en transmettant une primitive de demande TR-U-ABORT à la sous-couche transaction au point SP B.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
4.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION C: LE MESSAGE ABORT A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION D: LE DTID DANS LES MESSAGES CONTINUE ET ABORT ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
7.	VÉRIFICATION E: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

ABORT

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette d'information abandon par l'utilisateur: 01101011
Longueur d'information abandon par l'utilisateur: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.2.2 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.3.2.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté réponse; Réception IUT; Terminaison de base		
OBJET: Vérifier que le point sémaphore A peut terminer la transaction à la réception d'un message End succédant à un message Continue		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p>←————</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p>←————</p> <p><i>ind. TR-END</i></p> <p><=====</p>		<p>SP B (TSL)</p> <p>CONTINUE</p> <p>END</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B réponde par un message Continue.	
3.	Terminer la transaction par un message End (terminaison de base) émanant du point SP B.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
6.	VÉRIFICATION C: LE MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
7.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.2.2.2 2)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté réponse; Réception IUT; Abandon par la sous-couche transaction		
OBJET: Vérifier que le point sémaphore A peut terminer la transaction à la réception d'un message Abort (P) succédant à un message Continue		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p>←————</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p>←————</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>		<p>SP B (TSL)</p> <p>CONTINUE</p> <p>ABORT (P)</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B réponde par un message Continue.	
3.	Terminer la transaction par un message Abort (P) émanant du point SP B.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
6.	VÉRIFICATION C: LE MESSAGE ABORT A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
7.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: un octet
 Valeur de cause P-Abort: entier (0 à 4)

NUMÉRO DU TEST: 1.1.2.2.2.2 3)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2.1/Q.774 et 3.3.3.2.4/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue; Libération valide émanant du côté réponse; Réception IUT; Abandon par l'utilisateur-TR		
OBJET: Vérifier que le point sémaphore A peut terminer la transaction à la réception d'un message Abort (U) succédant à un message Continue		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p>←————</p> <p><i>ind. TR-CONTINUE</i></p> <p>←=====</p> <p>←————</p> <p><i>ind. TR-U-ABORT</i></p> <p>←=====</p>		<p>SP B (TSL)</p> <p>CONTINUE</p> <p>ABORT (U)</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B réponde par un message Continue.	
3.	Terminer la transaction par un message Abort (U) émanant du point SP B.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
6.	VÉRIFICATION C: LE MESSAGE ABORT A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
7.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

ABORT (U)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette d'information abandon par l'utilisateur: 01101011
 Longueur d'information abandon par l'utilisateur: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.3.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue (Partie composant non présente); Terminaison de base – Emission IUT		
OBJET: Vérifier que le point SP A est capable d'accepter un message Continue sans CP		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p>←————</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>dem. TR-END</i></p> <p>=====></p> <p>END</p> <p>—————→</p>		<p>SP B (TSL)</p> <p>CONTINUE</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie un message Continue sans CP au SP A.	
3.	Faire en sorte que le point SP A envoie un message End au point SP B.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
6.	VÉRIFICATION C: LE MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
7.	VÉRIFICATION D: L'AUTOMATE TSL A-T-IL ÉTÉ LAISSÉ À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets (1 à 4)
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets (1 à 4)
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets (1 à 4)
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets (1 à 4)
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message CONTINUE)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.3.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Libération après un message Continue (Partie composant non présente); Terminaison de base – Réception IUT		
OBJET: Vérifier que le point sémaphore A peut accepter un message Begin sans CP.		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>ind. TR-BEGIN</i> <=====	←-----	SP B (TSL) BEGIN
<i>dem. TR-CONTINUE</i> =====>	----->	
CONTINUE <i>ind. TR-END</i> <=====	-----> ←-----	END
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin sans CP au point SP A.	
2.	Faire en sorte que le point SP A envoie un message Continue au point SP B.	
3.	Faire en sorte que le point SP B envoie un message End sans CP au point SP A.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION C: LE MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
7.	VÉRIFICATION D: L'AUTOMATE TSL A-T-IL ÉTÉ LAISSÉ À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets (1 à 4)
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets (1 à 4)
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets (1 à 4)
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets (1 à 4)
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message CONTINUE)

NUMÉRO DU TEST: 1.1.2.4.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Echange de messages après l'établissement de la transaction; Lancement IUT		
OBJET: Vérifier le flux correct de messages entre le point SP A et le point SP B après l'établissement de la transaction (initialisée par l'IUT)		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>_____→</p> <p>_____←</p> <p><i>ind. TR-CONTINUE</i></p> <p>←=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====></p> <p>CONTINUE</p> <p>_____→</p> <p>_____←</p> <p><i>ind. TR-END</i></p> <p>←=====</p>		<p>SP B (TSL)</p> <p>CONTINUE</p> <p>END</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie un message Continue au point SP A.	
3.	Faire en sorte que le point SP A envoie un message Continue au point SP B.	
4.	Faire en sorte que le point SP B envoie un message END au point SP A.	
5.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
7.	VÉRIFICATION C: LE MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
8.	VÉRIFICATION D: LE MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
9.	VÉRIFICATION E: L'AUTOMATE TSL A-T-IL ÉTÉ LAISSÉ À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets (1 à 4)
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets (1 à 4)
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: 0 (longueur non valide)
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets (1 à 4)
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: 0 (longueur non valide)
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message CONTINUE)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

END

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: 0 (longueur non valide)
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.4.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.3/Q.774		
TITRE: Fonction valide; Dialogue structuré		
SOUS-TITRE: Echange de messages après l'établissement de la transaction; Réception IUT		
OBJET: Vérifier le flux correct de messages entre le point SP A et le point SP B après l'établissement de la transaction (réception IUT)		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>ind. TR-BEGIN</i> <===== <i>dem. TR-CONTINUE</i> =====> CONTINUE <i>ind. TR-CONTINUE</i> <===== <i>dem. TR-END</i> =====> END	 ←----- -----> ←----- ----->	SP B (TSL) BEGIN CONTINUE
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A.	
2.	Faire en sorte que le point SP A envoie un message Continue au point SP B.	
3.	Faire en sorte que le point SP B envoie un message Continue au point SP A.	
4.	Faire en sorte que le point SP A envoie un message END au point SP B.	
5.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
6.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
7.	VÉRIFICATION C: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
8.	VÉRIFICATION D: LE MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
9.	VÉRIFICATION E: L'AUTOMATE TSL A-T-IL ÉTÉ LAISSÉ À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Étiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Étiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets (longueur 1 à 4 octets)
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Étiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Étiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Étiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets (longueur 1 à 4 octets)
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Étiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: 0 (longueur non valide)
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Étiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Étiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Étiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets (longueur 1 à 4 octets)
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Étiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets (longueur 1 à 4 octets)
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message CONTINUE)
 Étiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

END

Étiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Étiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets (longueur 1 à 4 octets)
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Étiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.2.5.1		Feuillet: 1 de 1									
RÉFÉRENCE: 3.1.2.2.2.2/Q.771											
TITRE: Fonction valide, Dialogue structuré											
SOUS-TITRE: Adressage TC; Changement d'adresse de registre											
OBJET: Vérifier que le changement d'adresse de l'implémentation homologue correctement notifié a été enregistré et utilisé dans des messages ultérieurs.											
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos											
CONFIGURATION:1	TYPE DE TEST: VAT	TYPE DE SP: SP									
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; vertical-align: top;"> <p>SP A (TSL)</p> <p>BEGIN</p> </td> <td style="width: 40%; text-align: center; vertical-align: middle;"> <p>UDT (CLG A, CLD B)</p> <p>—————→</p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (TSL)</p> <p><i>ind. TR-BEGIN.</i></p> <p>=====→</p> <p><i>dem. TR-CONTINUE</i></p> <p>←=====</p> <p>CONTINUE</p> </td> </tr> <tr> <td style="vertical-align: top;"> <p><i>ind. TR-CONTINUE</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p>END</p> </td> <td style="text-align: center; vertical-align: middle;"> <p>UDT (CLG B*, CLD A)</p> <p>←—————</p> </td> <td></td> </tr> <tr> <td></td> <td style="text-align: center; vertical-align: middle;"> <p>UDT (CLG A, CLD B*)</p> <p>—————→</p> </td> <td></td> </tr> </table>			<p>SP A (TSL)</p> <p>BEGIN</p>	<p>UDT (CLG A, CLD B)</p> <p>—————→</p>	<p>SP B (TSL)</p> <p><i>ind. TR-BEGIN.</i></p> <p>=====→</p> <p><i>dem. TR-CONTINUE</i></p> <p>←=====</p> <p>CONTINUE</p>	<p><i>ind. TR-CONTINUE</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p>END</p>	<p>UDT (CLG B*, CLD A)</p> <p>←—————</p>			<p>UDT (CLG A, CLD B*)</p> <p>—————→</p>	
<p>SP A (TSL)</p> <p>BEGIN</p>	<p>UDT (CLG A, CLD B)</p> <p>—————→</p>	<p>SP B (TSL)</p> <p><i>ind. TR-BEGIN.</i></p> <p>=====→</p> <p><i>dem. TR-CONTINUE</i></p> <p>←=====</p> <p>CONTINUE</p>									
<p><i>ind. TR-CONTINUE</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p>END</p>	<p>UDT (CLG B*, CLD A)</p> <p>←—————</p>										
	<p>UDT (CLG A, CLD B*)</p> <p>—————→</p>										
DESCRIPTION DU TEST											
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.										
2.	Faire en sorte que le point SP B réponde avec un message Continue dans lequel l'adresse appelante a été optimisée.										
3.	Faire en sorte que le point SP A réponde avec un Message End.										
4.	VÉRIFICATION A: L'ADRESSE APPELÉE DANS L'EN-TÊTE DU MESSAGE SCCP POUR LA TERMINAISON ÉTAIT-ELLE LA MÊME QUE L'ADRESSE APPELANTE DANS L'EN-TÊTE DU MESSAGE SCCP POUR LE MESSAGE CONTINUE?										

NUMÉRO DU TEST: 1.1.3.1.1.1 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3/Q.774		
TITRE: Fonction Valide; Variations de codage et de valeur		
SOUS-TITRE: Variations de codage; Variations de longueur; Forme courte définie; Longueur de partie Composant sous forme courte définie intégrée en une forme courte		
OBJET: Vérifier que le point sémaphore A est capable d'accepter un message Begin dont la longueur est codée au moyen d'une forme courte définie et avec une partie composant dont la longueur est codée au moyen d'une forme courte définie.		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION:1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN.</i></p> <p><=====</p> <p><i>dem. TR-END</i></p> <p>=====></p> <p>(Terminaison de base)</p> <p>END</p>	<p>←</p> <p>→</p>	<p>SP B (TSL)</p> <p>BEGIN</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A avec des longueurs codées comme décrit dans l'objet du test.	
2.	Faire en sorte que le point SP A réponde avec un Message End.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A ET TRANSMIS À L'UTILISATEUR-TR?	
4.	VÉRIFICATION B: UN MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION C: LE DTID DANS LE MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Étiquette de type de message: 01100010

Longueur de type de message: nombre correct d'octets codés sous forme courte définie

Étiquette d'ID de transaction de départ: 01001000

Longueur d'ID de transaction de départ: un octet

Valeur d'ID de transaction de départ: chaîne d'octets (1 octet)

Étiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets codés sous forme courte définie

END

Étiquette de type de message: 01100100

Longueur de type de message: nombre correct d'octets

Étiquette d'ID de transaction de destination: 01001001

Longueur d'ID de transaction de destination: un octet

Valeur d'ID de transaction de destination: chaîne d'octets (1 octet)
(valeur de l'OTID reçue dans le message BEGIN)

Étiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.3.1.1.1 2)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3/Q.774		
TITRE: Fonction valide; Variations de codage et de valeur		
SOUS-TITRE: Variations de codage; Variations de longueur; Forme courte définie; Longueur de partie Composant sous forme courte définie intégrée dans la forme longue		
OBJET: Vérifier que le point sémaphore A peut accepter un message Begin dont la longueur est codée à l'aide de la forme longue définie et avec une partie Composant dont la longueur est codée à l'aide de la forme courte définie		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>ind. TR-BEGIN</i> <=====	←	SP B (TSL) BEGIN
<i>dem. TR-END</i> =====>		
(Terminaison de base) END	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A avec des longueurs codées comme indiqué à la rubrique "Objet" du test.	
2.	Faire en sorte que le point SP A réponde par un message End.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A ET TRANSMIS À L'UTILISATEUR-TR?	
4.	VÉRIFICATION B: UN MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION C: LE DTID DANS LE MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010

Longueur de type de message: nombre correct d'octets codés sous forme longue définie

Etiquette d'ID de transaction de départ: 01001000

Longueur d'ID de transaction de départ: un octet

Valeur d'ID de transaction de départ: chaîne d'octets (1 octet)

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets codés sous forme courte définie

END

Etiquette de type de message: 01100100

Longueur de type de message: nombre correct d'octets

Etiquette d'ID de transaction de destination: 01001001

Longueur d'ID de transaction de destination: un octet

Valeur d'ID de transaction de destination: chaîne d'octets (1 octet)
(valeur de l'OTID reçue dans le message BEGIN)

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.3.1.1.2 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3/Q.774		
TITRE: Fonction valide; Variations de codage et de valeur		
SOUS-TITRE: Variations de codage; Variations de longueur; Forme longue définie; Longueur de partie Composant sous forme longue définie intégrée dans la forme longue		
OBJET: Vérifier que le point sémaphore A peut accepter un message Begin dont la longueur est codée à l'aide de la forme longue définie et avec une partie Composant dont la longueur est codée à l'aide de la forme longue définie		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>ind. TR-BEGIN</i> <=====	←	SP B (TSL) BEGIN
<i>dem. TR-END</i> =====>		
(Terminaison de base) END	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A avec des longueurs codées comme indiqué à la rubrique "Objet" du test.	
2.	Faire en sorte que le point SP A réponde par un message End.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A ET TRANSMIS À L'UTILISATEUR-TR?	
4.	VÉRIFICATION B: UN MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION C: LE DTID DANS LE MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010

Longueur de type de message: nombre correct d'octets codés sous forme longue définie

Etiquette d'ID de transaction de départ: 01001000

Longueur d'ID de transaction de départ: un octet

Valeur d'ID de transaction de départ: chaîne d'octets (1 octet)

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets codés sous forme longue définie

END

Etiquette de type de message: 01100100

Longueur de type de message: nombre correct d'octets

Etiquette d'ID de transaction de destination: 01001001

Longueur d'ID de transaction de destination: un octet

Valeur d'ID de transaction de destination: chaîne d'octets (1 octet)
(valeur de l'OTID reçue dans le message BEGIN)

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.3.1.1.3 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3/Q.774		
TITRE: Fonction valide; Variations de codage et de valeur		
SOUS-TITRE: Variations de codage; Variations de longueur; Forme indéfinie; Longueur de partie Composant sous forme indéfinie intégrée dans la forme indéfinie		
OBJET: Vérifier que le point sémaphore A peut accepter un message Begin dont la longueur est codée à l'aide de la forme indéfinie et avec une partie Composant dont la longueur est codée à l'aide de la forme indéfinie		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p>(Terminaison de base)</p> <p>END</p>	<p>←-----</p> <p>BEGIN</p> <p>-----→</p>	<p>SP B (TSL)</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin au point SP A avec des longueurs codées comme indiqué à la rubrique "Objet" du test.	
2.	Faire en sorte que le point SP A réponde par un message End.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A ET TRANSMIS À L'UTILISATEUR-TR?	
4.	VÉRIFICATION B: UN MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION C: LE DTID DANS LE MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets codés sous forme indéfinie
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: un octet
Valeur d'ID de transaction de départ: chaîne d'octets (1 octet)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets codés sous forme indéfinie
Contenu du composant fourni par l'utilisateur TC
Etiquette EOC: 00000000, Longueur: 00000000

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: un octet
Valeur d'ID de transaction de destination: chaîne d'octets (1 octet)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.3.2.1 1)		Feuillet: 1 de 2
RÉFÉRENCE: 5.3/Q.774		
TITRE: Fonction valide; Variations de codage et de valeur		
SOUS-TITRE: Variations de valeur; ID de transaction; Longueur d'un octet		
OBJET: Vérifier que le point sémaphore A peut traiter un codage correct de l'élément d'information OTID (long d'un octet)		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>ind. TR-BEGIN</i> <===== <i>dem. TR-END</i> =====> (Terminaison de base) END	← →	SP B (TSL) BEGIN
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin, avec un OTID long d'un octet, au point SP A.	
2.	Faire en sorte que le point SP A réponde par un message End.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A ET TRANSMIS À L'UTILISATEUR-TR?	
4.	VÉRIFICATION B: UN MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION C: LE DTID DANS LE MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: un octet
Valeur d'ID de transaction de départ: chaîne d'octets (1 octet)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
longueur d'ID de transaction de destination: un octet
Valeur d'ID de transaction de destination: chaîne d'octets (1 octet)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.1.3.2.1 2)		Feuillet: 1 de 2
RÉFÉRENCE: 5.3/Q.774		
TITRE: Fonction valide; Variations de codage et de valeur		
SOUS-TITRE: Variations de valeur; ID de transaction; Longueur de quatre octets		
OBJET: Vérifier que le point sémaphore A peut traiter un codage correct de l'élément d'information OTID (long de 4 octets)		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>ind. TR-BEGIN</i> <===== <i>dem. TR-END</i> =====> (Terminaison de base) END	← →	SP B (TSL) BEGIN
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin, avec un OTID long de quatre octets, au point SP A.	
2.	Faire en sorte que le point SP A réponde par un message End.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A ET TRANSMIS À L'UTILISATEUR-TR?	
4.	VÉRIFICATION B: UN MESSAGE END A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION C: LE DTID DANS LE MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE BEGIN?	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: quatre octets
Valeur d'ID de transaction de départ: chaîne d'octets (4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

END

Etiquette de type de message: 01100100
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: quatre octets
Valeur d'ID de transaction de destination: chaîne d'octets (4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.2.1.1 1)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Valeurs non valides pour les éléments d'information		
SOUS-TITRE: Type de message Begin; Longueur d'OTID = 0		
OBJET: Vérifier que, à la réception d'un message Begin erroné, le point sémaphore A peut ignorer le message		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que le message Begin contienne une longueur d'OTID égale à 0		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>Détection de l'erreur de syntaxe</i>	←	SP B (TSL) BEGIN
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie le message Begin erroné, avec une longueur d'OTID égale à 0 octet, au point SP A.	
2.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR N'A PAS ÉTÉ INFORMÉ DU MESSAGE BEGIN.	
3.	VÉRIFICATION B: AUCUN MESSAGE N'A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN		
Etiquette de type de message: 01100010		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de départ: 01001000		
Longueur d'ID de transaction de départ: 0		
Valeur d'ID de transaction de départ: non présente		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		

NUMÉRO DU TEST: 1.2.1.1 2)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Valeurs non valides pour les éléments d'information		
SOUS-TITRE: Type de message Begin; Longueur d'OTID > quatre octets		
OBJET: Vérifier que le point sémaphore A peut traiter un codage non valide de l'élément d'information OTID		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que le message Begin contienne un OTID d'une longueur supérieure à quatre octets		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>Détection de l'erreur de syntaxe</i>	←—————	SP B (TSL) BEGIN
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie le message BEGIN erroné, avec un OTID long de 5 octets, au point SP A.	
2.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
3.	VÉRIFICATION B: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DE CET ÉVÉNEMENT.	
4.	VÉRIFICATION C: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE BEGIN.	
5.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN		
Etiquette de type de message: 01100010		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de départ: 01001000		
Longueur d'ID de transaction de départ: cinq		
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 5 octets)		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		

NUMÉRO DU TEST: 1.2.1.2 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Valeurs non valides pour les éléments d'information		
SOUS-TITRE: Premier message Continue; Longueur de DTID = 0		
OBJET: Vérifier que, à la réception d'un message Continue erroné, avec un DTID d'une longueur égale à 0 octet, le point sémaphore A peut ignorer le message ou abandonner la transaction correctement		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que le premier message Continue contienne un DTID d'une longueur égale à 0		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (TSL) SP B (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN _____→</p> <p><i>Détection de l'erreur de syntaxe</i> ←_____ CONTINUE</p> <p>ABORT (P) (voir la Note) _____→</p>		
NOTE – Si le message Abort n'est pas envoyé, il s'agit peut-être d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie le message Continue erroné.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE CONTINUE AU POINT SP A.	
5.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION AVANT LE MESSAGE CONTINUE ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT "INITIALISATION ENVOYÉE"?	
6.	VÉRIFICATION D: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A AVEC UN DTID ET UNE VALEUR DE CAUSE P-ABORT CORRECTS?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (1 à 4 octets)
(valeur de l'OTID reçue dans le message CONTINUE)
Etiquette de cause P-Abort: 01001010
Longueur de cause P-Abort: nombre correct d'octets
Valeur de cause P-Abort: partie transaction incorrecte

NUMÉRO DU TEST: 1.2.1.3 1)		Feuillet: 1 de 2						
RÉFÉRENCE: 3.3.4/Q.774								
TITRE: Comportement syntaxiquement non valide; Valeurs non valides pour les éléments d'information								
SOUS-TITRE: Message Continue ultérieur; Longueur de partie Composant incorrecte								
OBJET: Vérifier que, à la réception d'un message Continue erroné, avec un OTID déductible et un DTID déductible et affecté, après l'établissement de la transaction, le point sémaphore A peut abandonner la transaction.								
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos								
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP						
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 35%; vertical-align: top;"> <p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p>—————→</p> <p>←————</p> <p>—————→</p> </td> <td style="width: 35%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</p> <p>CONTINUE</p> </td> </tr> <tr> <td style="vertical-align: top;"> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>Détection de l'erreur</i></p> <p>ABORT (P) (voir la Note)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p> </td> <td style="vertical-align: top; text-align: center;"> <p>←————</p> <p>—————→</p> </td> <td style="vertical-align: top;"> <p>CONTINUE</p> <p>CONTINUE</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p>	<p>—————→</p> <p>←————</p> <p>—————→</p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>CONTINUE</p>	<p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>Détection de l'erreur</i></p> <p>ABORT (P) (voir la Note)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>←————</p> <p>—————→</p>	<p>CONTINUE</p> <p>CONTINUE</p>
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p>	<p>—————→</p> <p>←————</p> <p>—————→</p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>CONTINUE</p>						
<p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>Détection de l'erreur</i></p> <p>ABORT (P) (voir la Note)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>←————</p> <p>—————→</p>	<p>CONTINUE</p> <p>CONTINUE</p>						
NOTE – Si le message Abort n'est pas envoyé, il s'agit peut-être d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.								
DESCRIPTION DU TEST								
<ol style="list-style-type: none"> 1. 2. 3. 4. 5. 6. 7. 	<ol style="list-style-type: none"> Envoi d'un message Begin du point SP A au point SP B. Faire en sorte que le point SP B envoie un message Continue correct au point SP A. Faire en sorte que le point SP B envoie un message Continue erroné au point SP A (longueur de Partie Composant (CP) incorrecte). VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A? VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A? VÉRIFICATION C: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A, AVEC UN DTID ET UNE VALEUR DE CAUSE P-ABORT. VÉRIFICATION D: SI LE MESSAGE ABORT A ÉTÉ ENVOYÉ, LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A? 							

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE (1^{er} message)

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE (2^e message)

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur de l'OTID reçue dans le message CONTINUE)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: nombre correct d'octets
 Valeur de cause P-Abort: partie transaction mal formatée 00000010

NUMÉRO DU TEST: 1.2.1.4 1)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Valeurs non valides pour les éléments d'information		
SOUS-TITRE: Type de message End; Longueur de DTID > 4 octets		
OBJET: Vérifier que, à la réception d'un message End erroné, le point SP A peut ignorer le message		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que la longueur du DTID du message End soit supérieure à quatre octets		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL) SP B (TSL) <i>dem. TR-BEGIN</i> →</p> <p>BEGIN → <i>Détection de l'erreur</i> ← END</p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie un message End erroné au point SP A (longueur de DTID non valide).	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE END.	
5.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION AVANT LE MESSAGE END ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT INITIALISATION ENVOYÉE?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
<p>BEGIN</p> <p>Etiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette d'ID de transaction de départ: 01001000 Longueur d'ID de transaction de départ: nombre correct d'octets Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>END</p> <p>Etiquette de type de message: 01100100 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette d'ID de transaction de destination: 01001001 Longueur d'ID de transaction de destination: 00000101 (longueur non valide) Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 5 octets) (valeur de l'OTID reçue dans le message BEGIN)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>		

NUMÉRO DU TEST: 1.2.1.5 1)		Feuillet: 1 de 2			
RÉFÉRENCE: 3.3.4/Q.774					
TITRE: Comportement syntaxiquement non valide; Valeurs non valides pour les éléments d'information					
SOUS-TITRE: Message Abort; Valeur de cause P-Abort (Abandon par le fournisseur) non valide					
OBJET: Vérifier que le point sémaphore A peut traiter un codage incorrect de l'élément d'information de cause P-Abort (valeur illégale)					
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles qu'un message Abort avec un DTID déductible et affecté contienne une erreur de syntaxe et soit envoyé au point SP A en réponse au message Begin					
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 35%; vertical-align: top;"> <p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p><i>Détection de l'erreur de syntaxe</i></p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p>—————→</p> <p>←————</p> </td> <td style="width: 35%; vertical-align: top;"> <p>SP B (TSL)</p> <p>ABORT (P)</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p><i>Détection de l'erreur de syntaxe</i></p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>—————→</p> <p>←————</p>	<p>SP B (TSL)</p> <p>ABORT (P)</p>
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p><i>Détection de l'erreur de syntaxe</i></p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>—————→</p> <p>←————</p>	<p>SP B (TSL)</p> <p>ABORT (P)</p>			
NOTE – L'envoi de l'indication TR-Abort dépend du mode de mise en œuvre utilisé.					
DESCRIPTION DU TEST					
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B et que le point SP B réponde par un message Abort erroné (valeur de cause P-Abort illégale).				
2.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?				
3.	VÉRIFICATION B: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE ABORT ERRONÉ.				
4.	VÉRIFICATION C: SI L'INDICATION TR-ABORT A ÉTÉ ENVOYÉE, LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?				

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de cause P-Abort: 01001010
Longueur de cause P-Abort: nombre correct d'octets
Valeur de cause P-Abort: entier (5 – valeur illégale pour ce champ)

NUMÉRO DU TEST: 1.2.1.5 2)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Valeurs non valides pour les éléments d'information		
SOUS-TITRE: Message Abort; Longueur de cause P-Abort incorrecte		
OBJET: Vérifier que, à la réception d'un message ABORT erroné, avec longueur de cause incorrecte, le point sémaphore A peut ignorer le message et aviser l'utilisateur local		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles qu'un message Abort avec un DTID déductible et affecté contienne une erreur de syntaxe et soit envoyé au point SP A en réponse au message Begin		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (TSL) SP B (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN —————></p> <p><i>Détection de l'erreur de syntaxe</i> ←———— ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p>=====<</p>		
NOTE – L'envoi de l'indication TR-Abort dépend du mode de mise en œuvre utilisé.		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B et que le point SP B réponde par un message Abort erroné (longueur de cause P-Abort erronée).	
2.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
3.	VÉRIFICATION B: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE ABORT ERRONÉ.	
4.	VÉRIFICATION C: SI L'INDICATION TR-ABORT A ÉTÉ ENVOYÉE, LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur de l'OTID reçue dans le message BEGIN)
Etiquette de cause P-Abort: 01001010
Longueur de cause P-Abort: nombre correct d'octets (c'est-à-dire autre qu'une longueur d'un octet)
Valeur de cause P-Abort: entier (0 à 4)

NUMÉRO DU TEST: 1.2.2.1 1)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Type de message Unidirectional; Élément d'information inconnu présent		
OBJET: Vérifier que, à la réception d'un message Unidirectional erroné, le point sémaphore A peut ignorer le message		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles qu'un message Unidirectional contienne une erreur de syntaxe et soit envoyé au point SP A		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>Détection de l'erreur de syntaxe</i>	←	SP B (TSL) UNIDIRECTIONAL
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Unidirectional erroné au point SP A.	
2.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE UNIDIRECTIONAL AU POINT SP A.	
3.	VÉRIFICATION B: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ EN RÉPONSE AU MESSAGE UNIDIRECTIONAL.	
4.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
UNIDIRECTIONAL		
Etiquette de type de message: 01100001		
Longueur de type de message: nombre correct d'octets		
Partie composant manquante		

NUMÉRO DU TEST: 1.2.2.2 1)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Type de message Begin; OTID absent		
OBJET: Vérifier que, à la réception d'un message Begin erroné, le point sémaphore A peut ignorer le message		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que le message Begin contienne une erreur de syntaxe et que l'OTID ne soit pas déductible		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>Détection de l'erreur de syntaxe</i>	←	SP B (TSL) BEGIN
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie le message Begin erroné, avec OTID non présent, au point SP A.	
2.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DE CET ÉVÉNEMENT AU POINT SP A.	
3.	VÉRIFICATION B: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE SP A EN RÉPONSE AU MESSAGE BEGIN ERRONÉ.	
4.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN		
Etiquette de type de message: 01100010		
Longueur de type de message: nombre correct d'octets		
OTID absent		

NUMÉRO DU TEST: 1.2.2.2 2)		Feuillet: 1 de 2									
RÉFÉRENCE: 3.3.4/Q.774											
TITRE: Comportement syntaxiquement non valide; structure non valide											
SOUS-TITRE: Type de message Begin; Elément d'information inconnu présent											
OBJET: Vérifier que, à la réception d'un message Begin erroné, avec un élément d'information non valide, le point sémaphore A peut ignorer le message et créer un message Abort											
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que le message Begin contienne une erreur de syntaxe											
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP									
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%;">SP A (TSL)</td> <td style="width: 40%;"></td> <td style="width: 30%;">SP B (TSL)</td> </tr> <tr> <td><i>Détection de l'erreur de syntaxe</i></td> <td style="text-align: center;">←—————</td> <td>BEGIN</td> </tr> <tr> <td>ABORT (P)</td> <td style="text-align: center;">—————→</td> <td></td> </tr> </table>			SP A (TSL)		SP B (TSL)	<i>Détection de l'erreur de syntaxe</i>	←—————	BEGIN	ABORT (P)	—————→	
SP A (TSL)		SP B (TSL)									
<i>Détection de l'erreur de syntaxe</i>	←—————	BEGIN									
ABORT (P)	—————→										
NOTE – Si le message Abort n'est pas envoyé, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.											
DESCRIPTION DU TEST											
1.	Faire en sorte que le point SP B envoie le message Begin erroné, avec un élément d'information non valide après l'OTID, au point SP A.										
2.	VÉRIFICATION A: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT, AVEC DTID ET VALEUR DE CAUSE P-ABORT CORRECTS, DEPUIS LE POINT SP A?										
3.	VÉRIFICATION B: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?										

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'élément d'information: Inconnu (par exemple, 01101101)
Longueur d'élément d'information: nombre correct d'octets
Valeur d'élément d'information: chaîne d'octets

ABORT (P)

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur d'OTID reçue dans le message BEGIN)
Etiquette de cause P-Abort: 01001010
Longueur de cause P-Abort: nombre correct d'octets
Valeur de cause P-Abort: partie transaction incorrecte 00000011

NUMÉRO DU TEST: 1.2.2.3 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; structure non valide		
SOUS-TITRE: Premier message Continue; OTID absent		
OBJET: Vérifier que, à la réception d'un message Continue erroné, le point sémaphore A peut ignorer le message		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Le point SP B doit répondre par un message Continue à la réception du message Begin. Faire en sorte que les données au point SP B soient telles que le message Continue contienne une erreur de syntaxe et que l'OTID ne soit pas déductible		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Détection de l'erreur de syntaxe</i></p>	<p>—————></p> <p>←—————</p>	<p>SP B (TSL)</p> <p>CONTINUE</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie un message Continue erroné (OTID non déductible) au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A?	
4.	VÉRIFICATION B: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE CONTINUE AU POINT SP A.	
5.	VÉRIFICATION C: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE CONTINUE ERRONÉ.	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION AVANT LE MESSAGE CONTINUE ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT "INITIALISATION ENVOYÉE" AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets

OTID absent

Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur d'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.2.2.3 2)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; structure non valide		
SOUS-TITRE: Premier message Continue; DTID absent		
OBJET: Vérifier que, à la réception d'un message Continue erroné ne contenant pas de DTID, le point sémaphore A peut ignorer le message ou abandonner la transaction		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Le point SP B doit répondre par un message Continue à la réception du message Begin. Faire en sorte que les données au point SP B soient telles que le message Continue ne contienne pas de DTID		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Détection de l'erreur de syntaxe</i></p> <p>ABORT (P)</p>	<p>—————></p> <p>←————</p> <p>(DTID absent)</p> <p>—————></p>	<p>SP B (TSL)</p> <p>CONTINUE</p>
NOTE – Si le message Abort n'est pas envoyé, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie le message Continue erroné (DTID absent).	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A?	
4.	VÉRIFICATION B: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE CONTINUE AU POINT SP A.	
5.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION AVANT LE MESSAGE CONTINUE ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT "INITIALISATION ENVOYÉE"?	
6.	VÉRIFICATION D: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A AVEC UN DTID ET UNE VALEUR DE CAUSE P-ABORT CORRECTS?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)

DTID absent

Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message CONTINUE)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: nombre correct d'octets
 Valeur de cause P-Abort: partie transaction incorrecte 00000011

NUMÉRO DU TEST: 1.2.2.3 3)		Feuillet: 1 de 2			
RÉFÉRENCE: 3.3.4/Q.774					
TITRE: Comportement syntaxiquement non valide; structure non valide					
SOUS-TITRE: Premier message Continue; OTID dupliqué					
OBJET: Vérifier le comportement correct de l'implémentation sous test (IUT) à la réception d'un premier message Continue avec un OTID dupliqué					
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos					
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; vertical-align: top;"> <p>SP A (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p>ABORT (P) <i>ind. TR-P-ABORT</i> <=====</p> </td> <td style="width: 40%; vertical-align: middle; text-align: center;"> <p>—————→</p> <p>(avec OTID dupliqué)</p> <p>←—————</p> <p>—————→</p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</p> </td> </tr> </table>			<p>SP A (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p>ABORT (P) <i>ind. TR-P-ABORT</i> <=====</p>	<p>—————→</p> <p>(avec OTID dupliqué)</p> <p>←—————</p> <p>—————→</p>	<p>SP B (TSL)</p> <p>CONTINUE</p>
<p>SP A (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p>ABORT (P) <i>ind. TR-P-ABORT</i> <=====</p>	<p>—————→</p> <p>(avec OTID dupliqué)</p> <p>←—————</p> <p>—————→</p>	<p>SP B (TSL)</p> <p>CONTINUE</p>			
<p>NOTE – Si le message et la primitive ABORT n'ont pas été envoyés, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.</p>					
DESCRIPTION DU TEST					
1.	Faire en sorte que le point SP A envoie un message Begin.				
2.	Faire en sorte que le point SP B envoie un message Continue, avec un OTID dupliqué, au point SP A.				
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A?				
4.	VÉRIFICATION B: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A AVEC UNE VALEUR DE DTID ET UNE VALEUR DE CAUSE P-ABORT CORRECTES?				
5.	VÉRIFICATION C: SI LE MESSAGE ET LA PRIMITIVE ABORT ONT ÉTÉ ENVOYÉS, LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?				

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000 }
 Longueur d'ID de transaction de départ: nombre correct d'octets }
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) } Dupliqué
 Etiquette d'ID de transaction de départ: 01001000 }
 Longueur d'ID de transaction de départ: nombre correct d'octets }
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) }
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message CONTINUE)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: nombre correct d'octets
 Valeur de cause P-Abort: partie transaction incorrecte 00000011

NUMÉRO DU TEST: 1.2.2.3 4)		Feuillet: 1 de 2		
RÉFÉRENCE: 3.3.4/Q.774				
TITRE: Comportement syntaxiquement non valide; Structure non valide				
SOUS-TITRE: Premier message Continue; DTID dupliqué				
OBJET: Vérifier le comportement correct de l'implémentation sous test (IUT) à la réception d'un premier message Continue avec un DTID dupliqué				
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos				
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP		
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 35%; vertical-align: top;"> <p>SP A (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p style="text-align: center;">—————→</p> <p style="text-align: center;">(avec DTID dupliqué)</p> <p style="text-align: center;">←————</p> <p>ABORT (P) <i>ind. TR-P-ABORT</i> <=====</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p>SP B (TSL)</p> <p>CONTINUE</p> </td> </tr> </table>			<p>SP A (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p style="text-align: center;">—————→</p> <p style="text-align: center;">(avec DTID dupliqué)</p> <p style="text-align: center;">←————</p> <p>ABORT (P) <i>ind. TR-P-ABORT</i> <=====</p>	<p>SP B (TSL)</p> <p>CONTINUE</p>
<p>SP A (TSL) <i>dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p style="text-align: center;">—————→</p> <p style="text-align: center;">(avec DTID dupliqué)</p> <p style="text-align: center;">←————</p> <p>ABORT (P) <i>ind. TR-P-ABORT</i> <=====</p>	<p>SP B (TSL)</p> <p>CONTINUE</p>			
NOTE – Si le message et la primitive ABORT ne sont pas envoyés, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.				
DESCRIPTION DU TEST				
1.	Faire en sorte que le point SP A envoie un message Begin.			
2.	Faire en sorte que le point SP B envoie un message Continue, avec un DTID dupliqué, au point SP A.			
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A?			
4.	VÉRIFICATION B: UN MESSAGE ABORT AVEC VALEUR DE DTID ET VALEUR DE CAUSE ABORT CORRECTES A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A?			
5.	VÉRIFICATION C: SI LE MESSAGE ET LA PRIMITIVE ABORT ONT ÉTÉ ENVOYÉS, LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?			

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001 }
 Longueur d'ID de transaction de destination: nombre correct d'octets }
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) }
 (valeur d'OTID reçue dans le message BEGIN) }
 } Dupliqué
 Etiquette d'ID de transaction de destination: 01001001 }
 Longueur d'ID de transaction de destination: nombre correct d'octets }
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) }
 (valeur d'OTID reçue dans le message BEGIN) }
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message CONTINUE)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: nombre correct d'octets
 Valeur de cause P-Abort: partie transaction incorrecte 00000011

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message BEGIN)
 Etiquette d'élément d'information: inconnu (par exemple, 01101101)
 Longueur d'élément d'information: nombre correct d'octets
 Valeur d'élément d'information: CHAÎNE D'OCTETS

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message CONTINUE)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: nombre correct d'octets
 Valeur de cause P-Abort: partie transaction incorrecte 00000011

NUMÉRO DU TEST: 1.2.2.4 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Message Continue ultérieur; OTID absent		
OBJET: Vérifier que, à la réception d'un message Continue erroné après l'établissement de la transaction, le point sémaphore A peut ignorer le message		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>Détection de l'erreur</i></p>	<p>—————→</p> <p>←—————</p> <p>←—————</p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>CONTINUE</p>
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP A au point SP B.	
2.	Faire en sorte que le point SP B envoie un message Continue correct au point SP A.	
3.	Faire en sorte que le point SP B envoie un message Continue erroné au point SP A (OTID non déductible).	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT DU POINT SP A?	
5.	VÉRIFICATION B: LE PREMIER MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
6.	VÉRIFICATION C: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE CONTINUE ERRONÉ AU POINT SP A.	
7.	VÉRIFICATION D: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE CONTINUE ERRONÉ.	
8.	VÉRIFICATION E: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION AVANT LE MESSAGE CONTINUE ERRONÉ ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT ACTIF AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE (1^{er} message)

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur d'OTID reçue dans le message BEGIN)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE (2^e message)

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur d'OTID reçue dans le message BEGIN)

NUMÉRO DU TEST: 1.2.2.4 2)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Message Continue ultérieur; Elément d'information inconnu présent		
OBJET: Vérifier que, à la réception d'un message Continue erroné avec OTID déductible et DTID déductible et affecté, après l'établissement de la transaction, le point sémaphore A se comporte correctement		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>ind. TR-BEGIN</i></p> <p><=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====></p> <p>CONTINUE</p> <p><i>Détection de l'erreur</i></p> <p>ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>←-----</p> <p>-----></p> <p>←-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>CONTINUE</p>
NOTE – Si le message et la primitive ABORT ne sont pas envoyés, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.		
DESCRIPTION DU TEST		
1.	Envoi d'un message Begin du point SP B au point SP A.	
2.	Faire en sorte que le point SP A envoie un message Continue correct au point SP B.	
3.	Faire en sorte que le point SP B envoie un message Continue erroné au point SP A (élément d'information supplémentaire après l'élément d'information DTID).	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?	
5.	VÉRIFICATION B: LE PREMIER MESSAGE CONTINUE A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION C: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A AVEC UN DTID ET UNE VALEUR DE CAUSE P-ABORT CORRECTS?	
7.	VÉRIFICATION D: SI LE MESSAGE ABORT A ÉTÉ ENVOYÉ, LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE (1^{er} message)

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE (2^e message)

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID utilisée dans le message BEGIN)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message CONTINUE)
 Etiquette d'élément d'information: inconnu (par exemple, 01101101)
 Longueur d'élément d'information: nombre correct d'octets
 Valeur d'élément d'information: chaîne d'octets

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message BEGIN)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: nombre correct d'octets
 Valeur de cause P-Abort: partie transaction incorrecte 00000011

NUMÉRO DU TEST: 1.2.2.5 1)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Message End; DTID absent		
OBJET: Vérifier que, à la réception d'un message End erroné, le point sémaphore A peut ignorer le message		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que le message End contienne une erreur de syntaxe (DTID absent)		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL) SP B (TSL) <i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN _____></p> <p><i>Détection de l'erreur de syntaxe</i> ←_____ END</p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie un message End erroné (DTID absent) au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE AU POINT SP A.	
5.	VÉRIFICATION C: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE END ERRONÉ.	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION AVANT LE MESSAGE END ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT "INITIALISATION ENVOYÉE" AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN Etiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets Etiquette d'ID de transaction de départ: 01001000 Longueur d'ID de transaction de départ: nombre correct d'octets Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets END Etiquette de type de message: 01100100 Longueur de type de message: nombre correct d'octets DTID absent Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets		

NUMÉRO DU TEST: 1.2.2.6 1)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Message Abort; DTID absent		
OBJET: Vérifier que, à la réception d'un message Abort erroné, le point sémaphore A peut ignorer le message		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que le message Abort contienne une erreur de syntaxe (DTID absent)		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL) SP B (TSL) <i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN —————> <i>Détection de l'erreur de</i> ABORT <i>syntaxe</i> ←—————</p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie un message Abort erroné au point SP A.	
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
4.	VÉRIFICATION B: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE AU POINT SP A.	
5.	VÉRIFICATION C: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE ABORT ERRONÉ.	
6.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION AVANT MESSAGE ABORT ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT INITIALISATION ENVOYÉE AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN Etiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets Etiquette d'ID de transaction de départ: 01001000 Longueur d'ID de transaction de départ: nombre correct d'octets Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets ABORT (P) Etiquette de type de message: 01100111 Longueur de type de message: nombre correct d'octets DTID absent Etiquette de cause P-Abort: 01101100 Longueur de cause P-Abort: nombre correct d'octets Valeur de cause P-Abort: par exemple, partie transaction incorrecte 00000011		

NUMÉRO DU TEST: 1.2.2.7 1)		Feuillet: 1 de 1			
RÉFÉRENCE: 3.3.4/Q.774					
TITRE: Comportement syntaxiquement non valide; Structure non valide					
SOUS-TITRE: Message Unknown (inconnu); OTID non inclus					
OBJET: Vérifier que, à la réception d'un message Unknown, le point sémaphore A peut ignorer le message					
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles qu'un message Unknown avec un OTID non déductible soit envoyé au point SP A.					
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP			
SÉQUENCE DE MESSAGES ATTENDUE:					
<table style="width: 100%; border: none;"> <tr> <td style="width: 30%; vertical-align: top;"> SP A (TSL) <i>Détection du type de message Unknown</i> </td> <td style="width: 30%; text-align: center; vertical-align: middle;"> ←————— </td> <td style="width: 30%; vertical-align: top;"> SP B (TSL) MESSAGE UNKNOWN </td> </tr> </table>			SP A (TSL) <i>Détection du type de message Unknown</i>	←—————	SP B (TSL) MESSAGE UNKNOWN
SP A (TSL) <i>Détection du type de message Unknown</i>	←—————	SP B (TSL) MESSAGE UNKNOWN			
DESCRIPTION DU TEST					
1.	Faire en sorte que le point SP B envoie un message Unknown au point SP A.				
2.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DE CET ÉVÉNEMENT AU POINT SP A.				
3.	VÉRIFICATION B: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE UNKNOWN.				
4.	VÉRIFICATION C: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?				
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES					
MESSAGE UNKNOWN					
Etiquette de type de message: unknown (par exemple, 01100110)					
Longueur de type de message: nombre correct d'octets					
OTID absent					

NUMÉRO DU TEST: 1.2.2.7 2)		Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Message Unknown; OTID inclus et DTID non inclus		
OBJET: Vérifier que, à la réception d'un message Unknown, le point sémaphore A se comporte correctement		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles qu'un message Unknown avec un OTID déductible et un DTID non déductible ou déductible mais non affecté soit envoyé au SP A		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL) <i>Détection du type de message Unknown</i> ABORT (P)	← →	SP B (TSL) MESSAGE UNKNOWN
NOTE – Si le message Abort n'est pas envoyé, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Unknown au point SP A.	
2.	VÉRIFICATION A: SI UN MESSAGE P-ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A AVEC UN DTID ET UNE VALEUR DE CAUSE P-ABORT CORRECTS?	
3.	VÉRIFICATION B: SI LE MESSAGE ABORT A ÉTÉ ENVOYÉ, LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
MESSAGE UNKNOWN		
Etiquette de type de message: unknown (par exemple, 01100110)		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de départ: 01001000		
Longueur d'ID de transaction de départ: nombre correct d'octets		
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)		
ABORT (P)		
Etiquette de type de message: 01100111		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de destination: 01001001		
Longueur d'ID de transaction de destination: nombre correct d'octets		
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) (valeur d'OTID reçue dans le message UNKNOWN)		
Etiquette de cause P-Abort: 01001010		
Longueur de cause P-Abort: nombre correct d'octets		
Valeur de cause P-Abort: type de message non reconnu 00000000		

NUMÉRO DU TEST: 1.2.2.7 3)		Feuillet: 1 de 2			
RÉFÉRENCE: 3.3.4/Q.774					
TITRE: Comportement syntaxiquement non valide; Structure non valide					
SOUS-TITRE: Message Unknown; OTID inclus et DTID inclus					
OBJET: Vérifier que, à la réception d'un message Unknown avec DTID affecté, le point sémaphore A peut se comporter correctement					
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles qu'un message Unknown avec un OTID déductible et un DTID déductible et affecté soit envoyé au point SP A en réponse au message Begin					
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table border="0" style="width: 100%;"> <tr> <td style="width: 30%;"> <p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Détection du type de message Unknown</i></p> <p>ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p> </td> <td style="width: 40%; text-align: center;"> <p>—————></p> <p>←————</p> <p>—————></p> </td> <td style="width: 30%; vertical-align: middle;"> <p>SP B (TSL)</p> <p>MESSAGE UNKNOWN</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Détection du type de message Unknown</i></p> <p>ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>—————></p> <p>←————</p> <p>—————></p>	<p>SP B (TSL)</p> <p>MESSAGE UNKNOWN</p>
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Détection du type de message Unknown</i></p> <p>ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>—————></p> <p>←————</p> <p>—————></p>	<p>SP B (TSL)</p> <p>MESSAGE UNKNOWN</p>			
NOTE – Si le message et la primitive ABORT ne sont pas envoyés, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.					
DESCRIPTION DU TEST					
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B et que le point SP B réponde par le message Unknown.				
2.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?				
3.	VÉRIFICATION B: SI UN MESSAGE P-ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A AVEC UN DTID ET UNE VALEUR DE CAUSE P-ABORT CORRECTS?				
4.	VÉRIFICATION C: SI LE MESSAGE ABORT A ÉTÉ ENVOYÉ, L'UTILISATEUR-TR AU POINT SP A A-T-IL ÉTÉ AVISÉ PAR UNE PRIMITIVE D'INDICATION TR-P-ABORT QUE CETTE TRANSACTION AVAIT ÉTÉ ABANDONNÉE?				
5.	VÉRIFICATION D: SI LE MESSAGE ABORT A ÉTÉ ENVOYÉ, LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?				

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

MESSAGE UNKNOWN

Etiquette de type de message: unknown (par exemple, 01100110)
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message BEGIN)

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
 (valeur d'OTID reçue dans le message UNKNOWN)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: nombre correct d'octets
 Valeur de cause P-Abort: Type de message non reconnu 00000000

NUMÉRO DU TEST: 1.2.3.1 1)		Feuillet: 1 de 1									
RÉFÉRENCE: 3.3.4/Q.774											
TITRE: Comportement syntaxiquement non valide; Codage non valide											
SOUS-TITRE: Type de message Begin; Etiquette non valide											
OBJET: Vérifier que, à la réception d'un message Begin erroné, avec étiquette non valide, le point sémaphore A se comporte correctement											
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Faire en sorte que les données au point SP B soient telles que le message Begin contienne une étiquette non valide											
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP									
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%;">SP A (TSL)</td> <td style="width: 40%;"></td> <td style="width: 30%;">SP B (TSL)</td> </tr> <tr> <td><i>Détection de l'erreur de syntaxe</i></td> <td style="text-align: center;">←—————</td> <td>BEGIN</td> </tr> <tr> <td>ABORT (P)</td> <td style="text-align: center;">—————→</td> <td></td> </tr> </table>			SP A (TSL)		SP B (TSL)	<i>Détection de l'erreur de syntaxe</i>	←—————	BEGIN	ABORT (P)	—————→	
SP A (TSL)		SP B (TSL)									
<i>Détection de l'erreur de syntaxe</i>	←—————	BEGIN									
ABORT (P)	—————→										
NOTE – Si le message Abort n'est pas envoyé, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.											
DESCRIPTION DU TEST											
1.	Faire en sorte que le point SP B envoie le message Begin erroné au point SP A.										
2.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR N'A PAS ÉTÉ INFORMÉ DU MESSAGE BEGIN.										
3.	VÉRIFICATION B: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?										
4.	VÉRIFICATION C: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A AVEC UN DTID ET UNE VALEUR DE CAUSE P-ABORT CORRECTS?										
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES											
<p>BEGIN</p> <p>Etiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette non valide: par exemple, 00100010</p> <p>Etiquette d'ID de transaction de départ: 01001000 Longueur d'ID de transaction de départ: nombre correct d'octets Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)</p> <p>ABORT (P)</p> <p>Etiquette de type de message: 01100111 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette d'ID de transaction de destination: 01001001 Longueur d'ID de transaction de destination: nombre correct d'octets Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) (valeur d'OTID reçue dans le message BEGIN)</p> <p>Etiquette de cause P-Abort: 01001010 Longueur de cause P-Abort: nombre correct d'octets Valeur de cause P-Abort: partie transaction incorrecte 00000011</p>											

NUMÉRO DU TEST: 1.2.3.2 1)		Feuillet: 1 de 2			
RÉFÉRENCE: 3.3.4/Q.774					
TITRE: Comportement syntaxiquement non valide; Codage non valide					
SOUS-TITRE: Type de message Continue; Etiquette non valide					
OBJET: Vérifier que, à la réception d'un message Continue erroné avec étiquette non valide, le point sémaphore A se comporte correctement					
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos. Le point SP B doit répondre par un message Continue à la réception du message Begin. Faire en sorte que les données au point SP B soient telles que le message Continue contienne une erreur de syntaxe (étiquette non valide)					
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 35%; vertical-align: top;"> <p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Détection de l'erreur de syntaxe</i></p> <p>ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p>—————></p> <p><—————</p> <p>—————></p> </td> <td style="width: 35%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Détection de l'erreur de syntaxe</i></p> <p>ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>—————></p> <p><—————</p> <p>—————></p>	<p>SP B (TSL)</p> <p>CONTINUE</p>
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Détection de l'erreur de syntaxe</i></p> <p>ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p>	<p>—————></p> <p><—————</p> <p>—————></p>	<p>SP B (TSL)</p> <p>CONTINUE</p>			
NOTE – Si le message et la primitive ABORT ne sont pas envoyés, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.					
DESCRIPTION DU TEST					
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.				
2.	Faire en sorte que le point SP B envoie le message Continue erroné au point SP A.				
3.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?				
4.	VÉRIFICATION B: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE CONTINUE AU POINT SP A.				
5.	VÉRIFICATION C: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A AVEC UN DTID ET UNE VALEUR DE CAUSE P-ABORT CORRECTS?				

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette non valide: par exemple, 00011111
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur d'OTID reçue dans le message BEGIN)

ABORT (P)

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur d'OTID reçue dans le message CONTINUE)

Etiquette de cause P-Abort: 01001010
Longueur de cause P-Abort: nombre correct d'octets
Valeur de cause P-Abort: partie transaction incorrecte 00000011

NUMÉRO DU TEST: 1.3.1 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Messages inopportuns; Type de message Continue		
SOUS-TITRE: Réception d'un message Continue à l'état de repos avec DTID non affecté		
OBJET: Vérifier que, à la réception d'un message Continue avec un DTID non affecté, le point sémaphore A peut ignorer le message et créer un message Abort		
CONDITIONS INITIALES: Le point SP A (TSL) doit être à l'état de repos et le point SP B (TSL) doit être à l'état IR/actif. Faire en sorte que les données au point SP B soient telles qu'un message Continue avec un OTID déductible et un DTID déductible mais non affecté soit envoyé au point SP A		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (TSL)	←	SP B (TSL)
ABORT (P)	→	CONTINUE
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie le message Continue avec un DTID non affecté au point SP A.	
2.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE CONTINUE AU POINT SP A.	
3.	VÉRIFICATION B: LE DTID DANS LE MESSAGE ABORT ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE CONTINUE?	
4.	VÉRIFICATION C: UN MESSAGE ABORT A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A AVEC UNE VALEUR DE CAUSE P-ABORT D'UN ID DE TRANSACTION NON RECONNU?	
5.	VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

CONTINUE

Etiquette de type de message: 01100101
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de départ: 01001000
Longueur d'ID de transaction de départ: nombre correct d'octets
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
Etiquette de partie composant: 01101100
Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
Longueur de type de message: nombre correct d'octets
Etiquette d'ID de transaction de destination: 01001001
Longueur d'ID de transaction de destination: nombre correct d'octets
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets)
(valeur d'OTID reçue dans le message CONTINUE)
Etiquette de cause P-Abort: 01001010
Longueur de cause P-Abort: un octet
Valeur de cause P-Abort: ID de transaction non reconnu 00000001

NUMÉRO DU TEST: 1.4.1 1)		Feuillet: 1 de 2			
RÉFÉRENCE: 3.3.3.2/Q.774					
TITRE: Codage de transactions multiples; Codage de transaction valide					
SOUS-TITRE: Nouvelle demande de transaction pendant l'établissement de la transaction					
OBJET: Vérifier que le point sémaphore A peut réagir correctement à un message Begin pendant l'établissement d'une autre transaction					
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos					
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 35%; vertical-align: top;"> <p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>ind. TR-BEGIN</i></p> <p><=====</p> <p><i>dem. TR-END</i></p> <p>=====></p> <p>(Terminaison de base)</p> <p>(terminer nouvelle transaction)</p> <p>END</p> <p><i>ind. TR-END</i></p> <p><=====</p> </td> <td style="width: 30%; text-align: center; vertical-align: middle;"> <p>—————></p> <p>—————<</p> <p>—————></p> <p>—————<</p> </td> <td style="width: 35%; vertical-align: top;"> <p>SP B (TSL)</p> <p>BEGIN (nouvelle transaction)</p> <p>END</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>ind. TR-BEGIN</i></p> <p><=====</p> <p><i>dem. TR-END</i></p> <p>=====></p> <p>(Terminaison de base)</p> <p>(terminer nouvelle transaction)</p> <p>END</p> <p><i>ind. TR-END</i></p> <p><=====</p>	<p>—————></p> <p>—————<</p> <p>—————></p> <p>—————<</p>	<p>SP B (TSL)</p> <p>BEGIN (nouvelle transaction)</p> <p>END</p>
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>ind. TR-BEGIN</i></p> <p><=====</p> <p><i>dem. TR-END</i></p> <p>=====></p> <p>(Terminaison de base)</p> <p>(terminer nouvelle transaction)</p> <p>END</p> <p><i>ind. TR-END</i></p> <p><=====</p>	<p>—————></p> <p>—————<</p> <p>—————></p> <p>—————<</p>	<p>SP B (TSL)</p> <p>BEGIN (nouvelle transaction)</p> <p>END</p>			
DESCRIPTION DU TEST					
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.				
2.	Faire en sorte que le point SP B envoie un message Begin au point SP A (nouvelle transaction).				
3.	Faire en sorte que le point SP A réponde par un message End au deuxième message Begin.				
4.	Faire en sorte que le point SP B réponde par un message End au premier message Begin.				
5.	VÉRIFICATION A: LE PREMIER MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?				
6.	VÉRIFICATION B: LE DEUXIÈME MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?				
7.	VÉRIFICATION C: LE DTID DANS LE PREMIER MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE DEUXIÈME MESSAGE BEGIN?				
8.	VÉRIFICATION D: LE DEUXIÈME MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?				
9.	VÉRIFICATION E: LES AUTOMATES TSL ASSOCIÉS À CES TRANSACTIONS ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?				

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN (1^{er} message)

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) X
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

BEGIN (2^e message)

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) Y
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

END (1^{er} message)

Etiquette de type de message: 01100100
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) Y
 (valeur d'OTID reçue dans le 2^e message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

END (2^e message)

Etiquette de type de message: 01100100
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) X
 (valeur d'OTID reçue dans le 1^{er} message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.4.1 2)		Feuillet: 1 de 3
RÉFÉRENCE: 3.3.3.2/Q.774		
TITRE: Codage de transactions multiples; Codage de transaction valide		
SOUS-TITRE: Nouvelle demande de transaction après l'établissement de la transaction		
OBJET: Vérifier que le point sémaphore A peut réagir correctement à un message Begin après l'établissement d'une autre transaction		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>_____→</p> <p>←_____</p> <p><i>ind. TR-CONTINUE</i></p> <p>←=====</p> <p>_____←</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====></p> <p><i>(Terminaison de base)</i> <i>(Terminer nouvelle transaction)</i></p> <p>END</p> <p>_____→</p> <p>←_____</p> <p><i>ind. TR-END</i></p> <p>←=====</p>		<p>SP B (TSL)</p> <p>CONTINUE</p> <p>BEGIN (nouvelle transaction)</p> <p>END</p>

NUMÉRO DU TEST: 1.4.1 2)		Feuillet: 2 de 3
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B réponde par un message Continue au message Begin.	
3.	Faire en sorte que le point SP B envoie un message Begin au point SP A (nouvelle transaction).	
4.	Faire en sorte que le point SP A réponde par un message End au deuxième message Begin.	
5.	Faire en sorte que le point SP B réponde par un message End au premier message Begin.	
6.	VÉRIFICATION A: LE PREMIER MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
7.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?	
8.	VÉRIFICATION C: LE DEUXIÈME MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?	
9.	VÉRIFICATION D: LE DTID DANS LE PREMIER MESSAGE END ÉTAIT-IL IDENTIQUE À L'OTID DANS LE DEUXIÈME MESSAGE BEGIN?	
10.	VÉRIFICATION E: LE DEUXIÈME MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?	
11.	VÉRIFICATION F: LES AUTOMATES TSL ASSOCIÉS À CES TRANSACTIONS ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
BEGIN (1 ^{er} message)		
Etiquette de type de message: 01100010		
Longueur de type de message: nombre correct d'octets		
Etiquette d'ID de transaction de départ: 01001000		
Longueur d'ID de transaction de départ: nombre correct d'octets		
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) X		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) Y
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) X
 (valeur d'OTID reçue dans le 1^{er} message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

BEGIN (2^e message)

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) Z
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

END (1^{er} message)

Etiquette de type de message: 01100100
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) Z
 (valeur d'OTID reçue dans le 2^e message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

END (2^e message)

Etiquette de type de message: 01100100
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) X
 (valeur d'OTID reçue dans le 1^{er} message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.4.2 1)		Feuillet: 1 de 2
RÉFÉRENCE: 3.3.3.2/Q.774		
TITRE: Codage de transactions multiples; Messages inopportuns		
SOUS-TITRE: Message avec DTID non affecté pendant l'établissement de la transaction		
OBJET: Vérifier que le point sémaphore A peut réagir correctement à un message Continue avec DTID non affecté pendant l'établissement d'une autre transaction		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>—————→</p> <p>←————</p> <p>ABORT (P)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p> <p>←————</p> <p><i>ind. TR-END</i></p> <p><=====</p>		<p>SP B (TSL)</p> <p>CONTINUE (nouvelle transaction)</p> <p>END</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin au point SP B.	
2.	Faire en sorte que le point SP B envoie un message Continue avec DTID non affecté au point SP A.	
3.	Faire en sorte que le point SP B réponde par un message End au message BEGIN.	
4.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?	
5.	VÉRIFICATION B: LE MESSAGE CONTINUE A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?	
6.	VÉRIFICATION C: LE DTID DANS LE MESSAGE ABORT ÉTAIT-IL IDENTIQUE À L'OTID DANS LE MESSAGE CONTINUE?	
7.	VÉRIFICATION D: LA CAUSE P-ABORT DANS LE MESSAGE ABORT ÉTAIT-ELLE LA VALEUR CORRECTE (ID DE TRANSACTION NON RECONNU)?	
8.	VÉRIFICATION E: LE MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?	
9.	VÉRIFICATION F: LES AUTOMATES TSL ASSOCIÉS À CES TRANSACTIONS ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

BEGIN

Etiquette de type de message: 01100010
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) X
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

CONTINUE

Etiquette de type de message: 01100101
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de départ: 01001000
 Longueur d'ID de transaction de départ: nombre correct d'octets
 Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) Y
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) Z
 (non égale à X)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

ABORT (P)

Etiquette de type de message: 01100111
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) Y
 (valeur d'OTID reçue dans le message CONTINUE)
 Etiquette de cause P-Abort: 01001010
 Longueur de cause P-Abort: un octet
 Valeur de cause P-Abort: 00000001 ID de transaction non reconnu

END

Etiquette de type de message: 01100100
 Longueur de type de message: nombre correct d'octets
 Etiquette d'ID de transaction de destination: 01001001
 Longueur d'ID de transaction de destination: nombre correct d'octets
 Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) X
 (valeur d'OTID reçue dans le message BEGIN)
 Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

NUMÉRO DU TEST: 1.4.2 2)		Feuillet: 1 de 3
RÉFÉRENCE: 3.3.3.2/Q.774		
TITRE: Codage de transactions multiples; Messages inopportuns		
SOUS-TITRE: Message avec DTID non affecté après l'établissement de la transaction		
OBJET: Vérifier que le point sémaphore A peut réagir correctement à un message Continue avec DTID non affecté après l'établissement d'une autre transaction		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>_____→</p> <p>←_____</p> <p><i>ind. TR-CONTINUE</i></p> <p>←=====</p> <p>_____←</p> <p>CONTINUE</p> <p>_____→</p> <p><i>ind. TR-P-ABORT</i></p> <p>←=====</p> <p>_____←</p> <p>CONTINUE (nouvelle transaction)</p> <p>_____→</p> <p><i>ind. TR-END</i></p> <p>←=====</p>		<p>SP B (TSL)</p> <p>CONTINUE</p> <p>CONTINUE (nouvelle transaction)</p> <p>END</p>

NUMÉRO DU TEST: 1.4.2 2)	Feuillet: 2 de 3
DESCRIPTION DU TEST	
<ol style="list-style-type: none"> 1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 	<p>Faire en sorte que le point SP A envoie un message Begin au point SP B.</p> <p>Faire en sorte que le point SP B envoie un message Continue en réponse au message Begin émanant du point SP A.</p> <p>Faire en sorte que le point SP B envoie un message Continue avec un DTID non affecté au point SP A.</p> <p>Faire en sorte que le point SP B réponde par un message End au message Begin.</p> <p>VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?</p> <p>VÉRIFICATION B: LES MESSAGES CONTINUE ONT-ILS ÉTÉ REÇUS CORRECTEMENT PAR LE POINT SP A?</p> <p>VÉRIFICATION C: LE DTID DANS LE MESSAGE ABORT ÉTAIT-IL IDENTIQUE À L'OTID DANS LE DEUXIÈME MESSAGE CONTINUE?</p> <p>VÉRIFICATION D: LA CAUSE P-ABORT DANS LE MESSAGE ABORT ÉTAIT-ELLE LA VALEUR CORRECTE (ID DE TRANSACTION NON RECONNU)?</p> <p>VÉRIFICATION E: LE MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?</p> <p>VÉRIFICATION F: LES AUTOMATES TSL ASSOCIÉS À CES TRANSACTIONS ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?</p>
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES	
<p>BEGIN</p> <p>Etiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette d'ID de transaction de départ: 01001000 Longueur d'ID de transaction de départ: nombre correct d'octets Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) W</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>CONTINUE (1^{er} message)</p> <p>Etiquette de type de message: 01100101 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette d'ID de transaction de départ: 01001000 Longueur d'ID de transaction de départ: nombre correct d'octets Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) X</p> <p>Etiquette d'ID de transaction de destination: 01001001 Longueur d'ID de transaction de destination: nombre correct d'octets Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) W (valeur d'OTID reçue dans le message BEGIN)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>	

NUMÉRO DU TEST: 1.4.2 2)	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES	
CONTINUE (2 ^e message)	
Etiquette de type de message: 01100101	
Longueur de type de message: nombre correct d'octets	
Etiquette d'ID de transaction de départ: 01001000	
Longueur d'ID de transaction de départ: nombre correct d'octets	
Valeur d'ID de transaction de départ: chaîne d'octets (longueur: 1 à 4 octets) Y	
Etiquette d'ID de transaction de destination: 01001001	
Longueur d'ID de transaction de destination: nombre correct d'octets	
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) Z (non égale à W)	
Etiquette de partie composant: 01101100	
Longueur de partie composant: nombre correct d'octets	
ABORT (P)	
Etiquette de type de message: 01100111	
Longueur de type de message: nombre correct d'octets	
Etiquette d'ID de transaction de destination: 01001001	
Longueur d'ID de transaction de destination: nombre correct d'octets	
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) Y (valeur d'OTID reçue dans le 2 ^e message CONTINUE)	
Etiquette de cause P-Abort: 01001010	
Longueur de cause P-Abort: un octet	
Valeur de cause P-Abort: 00000001 ID de transaction non reconnu	
END	
Etiquette de type de message: 01100100	
Longueur de type de message: nombre correct d'octets	
Etiquette d'ID de transaction de destination: 01001001	
Longueur d'ID de transaction de destination: nombre correct d'octets	
Valeur d'ID de transaction de destination: chaîne d'octets (longueur: 1 à 4 octets) W (valeur d'OTID reçue dans le message BEGIN)	
Etiquette de partie composant: 01101100	
Longueur de partie composant: nombre correct d'octets	

7.2 Spécification de tests de la sous-couche composant TC (CSL)

7.2.1 Directives relatives à l'exécution des tests de la sous-couche composant

- a) pour tous les tests, la phrase "... composant avec une information correcte" dans la description du test signifie que les valeurs de détail dans le composant indiqué seront syntaxiquement vérifiées par rapport aux informations énumérées dans la table de vérification des composants contenus dans les messages;
- b) pour certains tests, il est nécessaire de vérifier que l'automate d'invocation revient à l'état de repos. A cet effet, il est possible d'envoyer un composant Retour Résultat-Dernier avec l'identificateur d'invocation présumé au repos. Si l'IUT (implémentation sous test) renvoie un Rejet avec un code de problème = "identificateur d'invocation non reconnu," l'IUT a réussi cette vérification;

- c) pour tous les tests de sous-couche CSL, le composant doit être transmis dans un message TSL, par exemple le composant d'Invocation dans l'essai 2.1.1.1 est transmis par le point SP A au point SP B dans un message Begin et le composant RetourRésultat-Dernier est transmis dans un Message End. En fait, si une transaction est d'abord établie entre le point SP A et le point SP B, il est possible de transmettre les composants d'Invocation et de Retour Résultat dans des messages de continuation;
- d) pour ces tests de sous-couche CSL, on suppose que la transaction reste active jusqu'à ce que le dernier composant dans le flux du message ait été fourni à son homologue. Lorsque cette hypothèse ne porte pas sur une application réelle (du fait par exemple de l'utilisation d'un message Abort ou End), il n'est pas possible de tirer des conclusions du test;
- e) les tests CSL supposent que le TSL et le SCCP fonctionnent correctement, et plus particulièrement que les composants sont transmis dans des messages TSL valides dans des états de transaction valide afin d'éviter toute situation anormale dans la ou les sous-couches sous-jacentes;
- f) les informations relatives à l'utilisateur du TC, telles que des opérations spécifiques aux codes et aux paramètres, ne sont pas spécifiées. C'est aux réalisateurs de tests d'inclure, le cas échéant, des informations dépendant de l'application, pour obtenir le flux du composant attendu;
- g) pour les tests de la partie dialogue, la table de vérification fournit ou non une "version du protocole". Dans les recommandations principales cette information est optionnelle et si elle n'est pas présente, la valeur par défaut est la "version 1".

7.2.2 Liste de tests de la sous-couche composant

Tous les tests sont des tests de validation

Les tests marqués d'un "*" sont des tests de compatibilité

2 Sous-couche composant

2.1 Fonctions valides

2.1.1 Composant Invoke, opérations non liées

2.1.1.1 Invocation d'opération simple de Classe 1

- * 2.1.1.1.1 IUT émetteur: réception résultat
- * 2.1.1.1.2 IUT récepteur: notification résultat
- * 2.1.1.1.3 IUT émetteur: réception d'erreur
- * 2.1.1.1.4 IUT récepteur: notification erreur
- * 2.1.1.1.5 IUT émetteur: expiration temporisateur

2.1.1.2 Invocation d'opération simple de Classe 2

- * 2.1.1.2.1 IUT émetteur: réception d'erreur
- * 2.1.1.2.2 IUT émetteur: expiration temporisateur

2.1.1.3 Invocation d'opération simple de Classe 3

- * 2.1.1.3.1 IUT émetteur: réception résultat
- * 2.1.1.3.2 IUT émetteur: expiration temporisateur

2.1.1.4 Invocation d'opération individuelle de Classe 4

- * 2.1.1.4.1 IUT émetteur

2.1.2 Composant Invoke, opérations liées

2.1.2.1 Invocation d'opération initiale de Classe 1

- * 2.1.2.1.1 IUT émetteur: réception d'une invocation d'opération corrélée de Classe 1, notification résultat

- * 2.1.2.1.2 IUT récepteur: émission d'une invocation d'opération liée de Classe 1, réception résultat
- * 2.1.2.1.3 IUT émetteur: réception d'une invocation d'opération liée de Classe 1, notification erreur
- * 2.1.2.1.4 IUT récepteur: envoi d'une invocation d'opération liée de Classe 1, réception erreur
- 2.1.2.2 Invocation d'opération initiale de Classe 4
- * 2.1.2.2.1 IUT émetteur: réception d'une invocation d'opération liée de Classe 2, pas de résultat
- * 2.1.2.2.2 IUT récepteur: envoi d'une invocation d'opération liée de Classe 2, expiration du temporisateur
- 2.1.3 Rejet distant
 - 2.1.3.1 Rejet distant par la sous-couche composant
 - 2.1.3.1.1 Code de problème général
 - 2.1.3.1.2 Code de problème Invoke
 - 2.1.3.1.3 Code de problème Return Result
 - 2.1.3.1.4 Code de problème Return Error
 - 2.1.3.2 Rejet distant par l'utilisateur-TC
 - 2.1.3.2.1 Code de problème Invoke
 - 2.1.3.2.2 Code de problème Return Result
 - 2.1.3.2.3 Code de problème Return Error
 - 2.1.3.3 Rejet distant avec un code de problème Invoke
 - 2.1.3.3.1 Invocation d'opération de Classe 1
 - 2.1.3.3.2 Invocation d'opération de Classe 2
 - 2.1.3.3.3 Invocation d'opération de Classe 3
 - 2.1.3.3.4 Invocation d'opération de Classe 4
- 2.1.4 Réception d'un composant entraînant un rejet par l'utilisateur-TC
 - 2.1.4.1 Problème Invoke
 - 2.1.4.1.1 Code d'opération non reconnu
 - 2.1.4.1.2 Opération liée non prévue
 - 2.1.4.1.3 Réponse liée non prévue
 - 2.1.4.1.4 Paramètre type erroné
 - 2.1.4.2 Problème Return Result
 - 2.1.4.2.1 Paramètre type erroné
 - 2.1.4.3 Problème Return Error
 - 2.1.4.3.1 Erreur non reconnue
 - 2.1.4.3.2 Erreur non prévue
 - 2.1.4.3.3 Paramètre type erroné
- 2.1.5 Segmentation du composant Return Result
 - 2.1.5.1 Invocation d'opération simple de Classe 1
 - * 2.1.5.1.1 IUT émetteur: réception des composants segmentés
 - * 2.1.5.1.2 IUT récepteur: émission des composants segmentés
 - 2.1.5.2 Invocation d'opération simple de Classe 3
 - * 2.1.5.2.1 IUT émetteur: réception des composants segmentés
- 2.1.6 Annulation par l'utilisateur
- 2.1.7 Variations de codage
 - 2.1.7.1 Longueur de composant: courte définie
 - 2.1.7.2 Longueur de composant: longue définie

- 2.1.7.3 Longueur de composant: indéfinie
- 2.1.7.4 Variations de valeur
 - 2.1.7.4.1 Identificateur Invoke
 - 2.1.7.4.1.1 Identificateur Invoke = -127 (FFh)
 - 2.1.7.4.1.2 Identificateur Invoke = 0 (00h)
 - 2.1.7.4.2 Code d'opération global
- 2.1.8 Groupement de composants multiples
 - 2.1.8.1 Invocation d'opérations multiples; réception succès
 - 2.1.8.2 Invocation d'opérations multiples; notification succès
 - 2.1.8.3 Composant mal formé reçu
- 2.1.9 Partie dialogue
 - 2.1.9.1 Accepter la proposition de contexte d'application
 - 2.1.9.1.1 Envoyer AARQ dans le message Begin
 - 2.1.9.1.2 Accepter AARQ et continuer le dialogue
 - 2.1.9.1.3 Accepter AARQ et terminer le dialogue
 - 2.1.9.2 Proposer un contexte d'application différent
 - 2.1.9.2.1 Envoyer APDU AARE avec une proposition alternative
 - 2.1.9.2.2 Recevoir APDU AARE avec la proposition alternative
 - 2.1.9.3 Dialogue refusé
 - 2.1.9.4 Dialogue abandonné
 - 2.1.9.5 Transport de l'information d'utilisateur
 - 2.1.9.5.1 Accepter l'information d'utilisateur dans un message Begin
 - 2.1.9.5.2 Accepter l'information d'utilisateur dans le premier message Continue
 - 2.1.9.5.3 Accepter l'information d'utilisateur dans un Message Continue ultérieur
 - 2.1.9.5.4 Accepter plusieurs éléments d'information d'utilisateur dans un message Continue
 - 2.1.9.6 Dialogue non structuré
 - 2.1.9.7 Version unité APDU de commande de dialogue
 - 2.1.9.7.1 Dialogue structuré, Version différente de 1
 - 2.1.9.7.2 Dialogue structuré, Version 1
 - 2.1.9.7.3 Dialogue non structuré, Version différente de 1
 - 2.1.9.7.4 Dialogue non structuré, Version 1
- 2.2 Comportement syntaxiquement non valide
 - 2.2.1 Valeurs non valides pour des éléments d'information
 - 2.2.1.1 Longueur d'identificateur d'invocation > 1 dans un composant d'invocation
 - 2.2.1.2 Longueur de l'identificateur Invoke égale à 0 dans le composant Invoke
 - 2.2.2 Structure non valide
 - 2.2.2.1 Composant Invoke
 - 2.2.2.1.1 Identificateur Invoke ID manquant
 - 2.2.2.1.2 Code d'opération manquant
 - 2.2.2.2 Composant Return Result
 - 2.2.2.2.1 Identificateur Invoke ID manquant
 - 2.2.2.2.2 Code d'opération manquant mais paramètres inclus
 - 2.2.2.2.3 Étiquette de séquence manquante mais paramètres inclus
 - 2.2.2.3 Return Error

- 2.2.2.3.1 Identificateur Invoke ID manquant
- 2.2.2.3.2 Code d'erreur manquant
- 2.2.2.4 Type de composant Unknown
 - 2.2.2.4.1 Identificateur Invoke ID non reconnaissable
 - 2.2.2.4.2 Identificateur Invoke ID déductible
- 2.2.2.5 Partie dialogue
 - 2.2.2.5.1 Contexte d'application manquant dans APDU AARQ
 - 2.2.2.5.2 Longueur incorrecte
 - 2.2.2.5.3 Diagnostic de l'origine du résultat manquant
 - 2.2.2.5.4 Contexte d'application manquant dans APDU AUDT
 - 2.2.2.5.5 Type externe sans référence directe
 - 2.2.2.5.6 Référence indirecte dans un type externe
 - 2.2.2.5.7 Information d'utilisateur sans référence directe
 - 2.2.2.5.8 Référence indirecte dans une information d'utilisateur
- 2.2.3 Codage non valide pour composant Invoke
 - 2.2.3.1 Étiquette non valide
 - 2.2.3.2 Longueur de composant erroné
 - 2.2.3.3 EOC manquant dans la forme indéfinie
- 2.3 Comportement inopportun
 - 2.3.1 Composant Invoke inopportun
 - 2.3.1.1 ID lié non valide
 - 2.3.2 ID Invoke non reconnu
 - 2.3.2.1 Composant Return Result-Last inopportun
 - 2.3.2.2 Composant Return Result-Not-Last inopportun
 - 2.3.2.3 Composant Return Error inopportun
 - 2.3.2.4 Composant Reject inopportun
 - 2.3.3 Composants imprévus
 - 2.3.3.1 Return Result-Last pour Classe 2
 - 2.3.3.2 Return Result-Last pour Classe 4
 - 2.3.3.3 Return Result Not-Last pour Classe 2
 - 2.3.3.4 Return Result Not-Last pour Classe 4
 - 2.3.3.5 Return Error pour Classe 3
 - 2.3.3.6 Return Error pour Classe 4
 - 2.3.4 Partie dialogue, unités APDU imprévues
 - 2.3.4.1 Message Begin avec une APDU AARE
 - 2.3.4.2 Confirmation de dialogue avec une unité APDU autre que AARE
 - 2.3.4.3 Confirmation de dialogue avec une APDU ABRT
 - 2.3.4.4 Présence d'une partie dialogue APDU à l'état actif
 - 2.3.4.5 Message Unidirectional avec une syntaxe abstraite non attendue
 - 2.3.4.6 Partie dialogue non attendue dans le message Continue
 - 2.3.4.7 Partie dialogue manquante dans le message Continue
 - 2.3.4.8 Message Begin avec syntaxe abstraite non attendue

NUMÉRO DU TEST: 2.1.1.1.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 1; IUT émetteur: réception résultat		
OBJET: Vérifier qu'une opération simple de Classe 1 peut être invoquée avec succès et que l'indication d'exécution avec succès de l'opération peut être reçue et transmise à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un Composant Return Result-Last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-RESULT-L</i></p> <p>=====<</p>	<p>SP B (CSL)</p> <p>—————></p> <p>—————<</p> <p>RETURN RESULT-LAST (i)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant Invoke dans un messageTSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC)

COMPOSANT RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de Composant: 10100010 (RETURN RESULT-LAST)
Longueur de composant: nombre correct d'octets
Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i
Etiquette de séquence: 00110000 (voir la Note)
Longueur de séquence: nombre correct d'octets (voir la Note)
Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
Code d'opération: x (voir la Note)
paramètres (fournis par l'utilisateur-TC)
NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.1.1.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 1; IUT récepteur: notification résultat		
OBJET: Vérifier qu'une opération de Classe 1 peut être invoquée avec succès et que l'indication d'exécution avec succès de l'opération peut être envoyée		
CONDITIONS INITIALES: Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ← INVOKE (i) ind. TC-INVOKE <===== dem. TC-RESULT-L =====> RETURN-RESULT-LAST (i) → </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'ID D'INVOCATION DANS LE COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
5.	VÉRIFICATION D: LE CODE D'OPÉRATION DANS LE COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans un message TSL du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)
Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.1.1.3		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 1; IUT émetteur: réception d'erreur		
OBJET: Vérifier qu'une opération de Classe 1 peut être invoquée avec succès et que l'indication de non-exécution de l'opération peut être reçue et transmise à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i) <i>ind. TC-U-ERROR</i> <=====	—————→ ←—————	 RETURN ERROR (i)
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN ERROR AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans un message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
Code d'erreur: y (y est un code d'erreur valide)
paramètres (fournis par l'utilisateur-TC)

NUMÉRO DU TEST: 2.1.1.1.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 1; IUT récepteur: notification d'erreur		
OBJET: Vérifier qu'une opération de Classe 1 peut être invoquée avec succès et que l'indication de non-exécution avec succès de l'opération peut être envoyée		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke 2) Faire en sorte que les données de l'utilisateur-TC au point SP A soient telles qu'un composant Return-Error puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>ind. TC-U-ERROR</i> <===== <i>dem. TC-RESULT-L</i> =====> RETURN-ERROR (i)	← →	SP B (CSL) INVOKE (i)
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN ERROR AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT RETURN ERROR ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
5.	VÉRIFICATION D: LE CODE D'ERREUR DANS LE COMPOSANT RETURN ERROR ÉTAIT-IL VALIDE?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans un message TSL du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC)

COMPOSANT RETURN ERROR dans le message TSL du point SP A au point SP B

Etiquette de type de Composant: 10100011 (RETURN ERROR)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
Code d'erreur: y

paramètres (fournis par l'utilisateur-TC)

NUMÉRO DU TEST: 2.1.1.1.5		Feuille: 1 de 1
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 1; IUT émetteur: expiration temporisateur		
OBJET: Vérifier qu'une opération de Classe 1 peut être invoquée avec succès et que l'indication d'expiration de temporisateur peut être transmise à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP A soient telles qu'aucun composant ne puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) —————></p> <p>expiration du temporisateur pour l'invocation (i)</p> <p><i>ind. TC-L-CANCEL (i)</i></p> <p><=====</p>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE FLUX DE COMPOSANTS ÉTAIT-IL COMME INDIQUÉ CI-DESSUS?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans un messageTSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		
Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)		
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)		
Code d'opération: x (x représente un code d'opération valide)		
paramètres (fournis par l'utilisateur-TC)		

NUMÉRO DU TEST: 2.1.1.2.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 2; IUT émetteur: réception d'erreur		
OBJET: Vérifier qu'une opération de Classe 2 peut être invoquée avec succès et que l'indication de défaillance peut être reçue et transmise à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) —————→</p> <p>←———— RETURN ERROR (i)</p> <p><i>ind. TC-U-ERROR</i></p> <p>←=====</p>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 2 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN ERROR AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans un message TSL du point SP A au point SP B		
Etiquette de type de Composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC)

COMPOSANT RETURN ERROR dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)
Longueur de composant: nombre correct d'octets
Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i
Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
Code erreur: y
paramètres (fournis par l'utilisateur-TC)

NUMÉRO DU TEST: 2.1.1.2.2		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 2; IUT émetteur: expiration temporisateur		
OBJET: Vérifier qu'une opération de Classe 2 peut être invoquée avec succès et que l'indication d'expiration de temporisateur peut être transmise à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'aucun composant ne puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) —————></p> <p>expiration temporisateur pour d'invocation (i)</p> <p><i>ind. TC-L-CANCEL</i></p> <p><=====</p>		
DESCRIPTION DU TEST		
1. Invocation d'une opération simple de Classe 2 du point SP A au point SP B. 2. VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A? 3. VÉRIFICATION B: LE FLUX DE COMPOSANTS ÉTAIT-IL COMME INDIQUÉ CI-DESSUS? 4. VÉRIFICATION C: L'UTILISATEUR-TC AU POINT SP A A-T-IL ÉTÉ INFORMÉ DE L'EXPIRATION DU TEMPORISATEUR? 5. VÉRIFICATION D: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?		
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets Composant INVOKE dans un message TSL du point SP A au point SP B Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000001 (un octet) ID d'invocation: i (i représente un nombre entier) Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) Code d'opération: x (x représente un code d'opération valide) paramètres (fournis par l'utilisateur-TC)		

NUMÉRO DU TEST: 2.1.1.3.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 3; IUT émetteur: réception résultat		
OBJET: Vérifier qu'une opération simple de Classe 3 peut être invoquée avec succès et que l'indication d'exécution avec succès de l'opération peut être reçue et transmise à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-RESULT-L</i></p> <p>=====</p>	<p>—————></p> <p>←————</p>	<p>SP B (CSL)</p> <p>RETURN RESULT-LAST (i)</p>
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 3 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.1.3.2		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 3; IUT émetteur: expiration temporisateur		
OBJET: Vérifier qu'une opération de Classe 3 peut être invoquée avec succès et que l'indication d'expiration de temporisateur peut être transmise à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'aucun composant ne puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) —————></p> <p>expiration du temporisateur pour l'invocation (i)</p> <p><i>ind. TC-L-CANCEL</i></p> <p><=====</p>		
DESCRIPTION DU TEST		
1. Invocation d'une opération de Classe 3 du point SP A au point SP B. 2. VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL-ÉTÉ ENVOYÉ PAR LE POINT SP A? 3. VÉRIFICATION B: LE FLUX DE COMPOSANTS ÉTAIT-IL COMME INDIQUÉ CI-DESSUS? 4. VÉRIFICATION C: L'UTILISATEUR-TC AU POINT SP A A-T-IL ÉTÉ INFORMÉ DE L'EXPIRATION DU TEMPORISATEUR? 5. VÉRIFICATION D: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?		
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets Composant INVOKE dans le message TSL du point SP A au point SP B Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000001 (un octet) ID d'invocation: i (i représente un nombre entier) Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) Code d'opération: x (x représente un code d'opération valide) paramètres (fournis par l'utilisateur-TC)		

NUMÉRO DU TEST: 2.1.1.4.1		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non liées		
SOUS-TITRE: Invocation d'opération simple de Classe 4; IUT émetteur		
OBJET: Vérifier qu'une opération de Classe 4; peut être initialisée avec succès et qu'aucune réponse n'est reçue		
CONDITIONS INITIALES: Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <p>SP A (CSL) SP B (CSL) <i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) —————> expiration du temporisateur pour l'invocation (i) <i>ind. TC-L-CANCEL</i></p> <p>←=====</p>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 4 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: L'UTILISATEUR-TC AU POINT SP A A-T-IL ÉTÉ INFORMÉ DE L'EXPIRATION DU TEMPORISATEUR?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets Composant INVOKE dans le message TSL du point SP A au point SP B Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000001 (un octet) ID d'invocation: i (i représente un nombre entier) Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) Code d'opération: x (x représente un code d'opération valide) paramètres (fournis par l'utilisateur-TC)		

NUMÉRO DU TEST: 2.1.2.1.1		Feuillet: 1 de 3
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations liées		
SOUS-TITRE: Invocation d'opération initiale de Classe 1; IUT émetteur; réception d'une invocation d'opération liée de Classe 1; notification résultat.		
OBJET: Vérifier qu'une opération liée de Classe 1 peut être reçue avec succès et que l'indication d'exécution avec succès de l'opération initiale peut être reçue correctement.		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Invoke lié puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-INVOKE</i></p> <p><=====</p> <p><i>dem. TC-RESULT-L</i></p> <p>=====></p> <p>RETURN-RESULT-LAST (j)</p> <p><i>ind. TC-RESULT-L</i></p> <p><=====</p>	<p>SP B (CSL)</p> <p>-----></p> <p>INVOKE (j, i)</p> <p>-----<</p> <p>-----></p> <p>RETURN-RESULT-LAST (i)</p> <p>-----<</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération liée du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: UN COMPOSANT INVOKE LIÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
5.	VÉRIFICATION D: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE ENVOYÉ PAR LE POINT SP B?	
6.	VÉRIFICATION E: LE CODE D'OPÉRATION DANS LE COMPOSANT RETURN RESULT-LAST ENVOYÉ PAR LE POINT SP A ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE ENVOYÉ PAR LE POINT SP B?	
7.	VÉRIFICATION F: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
8.	VÉRIFICATION G: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie Composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyé par le point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: j (j représente un nombre entier)

Etiquette d'ID lié: 10000000

Longueur d'ID lié: 00000001 (un octet)

ID lié: i

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'opération: y (y représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le 2^e message TSL envoyé par le point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet) (voir la Note)

Code d'opération: y (voir la Note)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL envoyé du point SP B au point SP A

Etiquette de type de Composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.2.1.2		Feuillet: 1 de 3
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations liées		
SOUS-TITRE: Invocation d'opération initiale de Classe 1; IUT récepteur; émission d'une invocation d'opération liée de Classe 1, réception résultat		
OBJET: Vérifier qu'une opération liée de Classe 1 peut être invoquée avec succès et que l'opération initiale peut être exécutée avec succès		
CONDITIONS INITIALES: Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié crée au point SP B contienne un composant Invoke qui invoquera une opération liée		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ←----- INVOKE (i) ind. TC-INVOKE <----- dem. TC-INVOKE =====> INVOKE (j, i) -----> ←----- RETURN RESULT-LAST (j) ind. TC-RESULT-L <===== dem. TC-RESULT-L =====> RETURN-RESULT-LAST (i) -----> </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération liée du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT INVOKE LIÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'ID LIÉ ÉTAIT-IL IDENTIQUE À L'IDENTIFICATEUR INVOKE ID INITIAL ENVOYÉ PAR LE POINT SP B?	
5.	VÉRIFICATION D: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
6.	VÉRIFICATION E: LE DEUXIÈME COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
7.	VÉRIFICATION F: L'IDENTIFICATEUR INVOKE ID DANS LE DEUXIÈME COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE INITIAL ENVOYÉ PAR LE POINT SP B?	
8.	VÉRIFICATION G: LE CODE D'OPÉRATION DANS LE DEUXIÈME COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE INITIAL?	
9.	VÉRIFICATION H: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie Composant dans les messages TSL

Etiquette de partie composant: 01101100
 Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL initial du point SP B au point SP A

Etiquette de type de composant: 10100001 (INVOKE)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: i (i représente un nombre entier)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyé du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: j (j représente un nombre entier)
 Etiquette d'ID lié: 10000000
 Longueur d'ID lié: 00000001 (un octet)
 ID lié: i
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
 Code d'opération: y (y représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL envoyé par le point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: j

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple 00000001 si y est long d'un octet) (voir la Note)

Code d'opération: y (voir la Note)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL envoyé par le point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.2.1.3		Feuillet: 1 de 3
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations liées		
SOUS-TITRE: Invocation d'opération initiale de Classe 1; IUT émetteur; réception d'une invocation d'opération liée de Classe 1, notification erreur		
OBJET: Vérifier qu'une opération liée de Classe 1 peut être reçue avec succès et que la notification d'erreur n'aura pas d'incidence sur l'exécution avec succès de l'opération initiale		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'une opération liée puisse être créée		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANT ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-INVOKE</i></p> <p><=====</p> <p><i>dem. TC-U-ERROR</i></p> <p>=====></p> <p>RETURN-ERROR (j)</p> <p><i>ind. TC-RESULT-L</i></p> <p><=====</p>	<p>—————></p> <p>—————<</p> <p>—————></p> <p>—————<</p>	<p>SP B (CSL)</p> <p>INVOKE (j, i)</p> <p>RETURN-RESULT-LAST (i)</p>
DESCRIPTION DU TEST		
1.	Invocation d'une opération liée du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: UN COMPOSANT INVOKE LIÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: LE COMPOSANT RETURN ERROR AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
5.	VÉRIFICATION D: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT RETURN ERROR ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE ENVOYÉ PAR LE POINT SP B?	
6.	VÉRIFICATION E: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
7.	VÉRIFICATION F: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyés par le point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: j (j représente un nombre entier)

Etiquette d'ID lié: 10000000

Longueur d'ID lié: 00000001 (un octet)

ID lié: i

Etiquette de code l'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code l'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'opération: y (y représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL envoyés par le point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale) (voir la Note)
Longueur de code d'erreur: nombre correct d'octets (par exemple 00000001 si z est long d'un octet) (voir la Note)
Code d'erreur: z (voir la Note)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL envoyé par le point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.2.1.4		Feuillet: 1 de 3
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations liées		
SOUS-TITRE: Invocation d'opération initiale de Classe 1; IUT récepteur: envoi d'une invocation d'opération liée de Classe 1, réception erreur		
OBJET: Vérifier qu'une opération liée de Classe 1 peut être invoquée avec succès et que la réception d'une erreur n'aura pas d'incidence sur l'exécution avec succès de l'opération initiale		
CONDITIONS INITIALES: Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ←----- INVOKE (i) ind. TC-INVOKE <===== dem. TC-INVOKE =====> INVOKE (j, i) -----> ←----- RETURN ERROR (j) ind. TC-U-ERROR <===== dem. TC-RESULT-L =====> RETURN-RESULT-LAST (i) -----> </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération liée du point SP B au point SP A	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT INVOKE LIÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'ID LIÉ ÉTAIT-IL IDENTIQUE À L'IDENTIFICATEUR INVOKE ID INITIAL ENVOYÉ PAR LE POINT SP B?	
5.	VÉRIFICATION D: LE COMPOSANT RETURN ERROR AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
6.	VÉRIFICATION E: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
7.	VÉRIFICATION F: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE INITIAL ENVOYÉ PAR LE POINT SP B?	
8.	VÉRIFICATION G: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyés par le point SP A

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: j (j représente un nombre entier)

Etiquette d'ID lié: 10000000

Longueur d'ID lié: 00000001 (un octet)

ID lié: i

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple 00000001 si y est long d'un octet)

Code d'opération: y (y représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL envoyé par le point SP B

Etiquette de type de composant: 10100011

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si z est long d'un octet)
 code d'erreur: z

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL envoyé par le point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) or 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple 00000001 si x est long d'un octet)
 (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.2.2.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations liées		
SOUS-TITRE: Invocation d'opération initiale de Classe 4; IUT émetteur; réception d'une demande d'invocation d'opération liée de Classe 2, pas de résultat		
OBJET: Vérifier qu'une opération liée de Classe 2 peut être reçue avec succès et que l'opération initiale de Classe 4 peut être exécutée avec succès		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke de Classe 4		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Invoke lié de Classe 2 puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL) <i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) —————></p> <p><————— INVOKE (j, i)</p> <p><i>ind. TC-INVOKE</i></p> <p><===== expiration du temporisateur pour l'invocation (i)</p> <p><i>ind. TC-L-CANCEL</i></p> <p><===== </p>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération liée du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT INVOKE LIÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyé par le point SP B

Etiquette de type de composant: 10100001 (INVOKE)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: j (j représente un nombre entier)
 Etiquette d'ID lié: 10000000
 Longueur d'ID lié: 00000001 (un octet)
 ID lié: i
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
 Code d'opération: y (y représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyé du point SP A au point SP B

Etiquette de type de composant: 10100010 (INVOKE)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: j (j représente un nombre entier)

Etiquette d'ID lié: 10000000
 Longueur d'ID lié: 00000001 (un octet)
 ID lié: i

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
 Code d'opération: y (y représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

NUMÉRO DU TEST: 2.1.3.1.1		Feuillet: 1 de 2
RÉFÉRENCES: 3.2.1/Q.774, 3.7.1/Q.772		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant par la sous-couche composant; code de problème général		
OBJET: Vérifier q'un rejet distant par la sous-couche composant, avec code de problème général, peut être transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet puisse être créé avec code de problème général		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====> INVOKE (i)		SP B (CSL) REJECT (i)
<i>ind. TC-R-REJECT</i> <=====		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 1 au point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie Composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets
Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i
Etiquette de code de problème: 10000000 (problème général)
Longueur de code de problème: 00000001
Code de problème: 00000000 (composant non reconnu)

NUMÉRO DU TEST: 2.1.3.1.2		Feuillet: 1 de 2
RÉFÉRENCES: 3.2.1/Q.774; 3.7.2/Q.772		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant par la sous-couche composant; Code de problème Invoke		
OBJET: Vérifier qu'un rejet distant par la sous-couche composant avec code de problème Invoke peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet avec code de problème Invoke puisse être créé</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>ind. TC-INVOKE</i></p> <p><=====</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (j, i)</p> <p><i>ind. TC-R-REJECT</i></p> <p><=====</p>	<p>←</p> <p>→</p> <p>←</p>	<p>SP B (CSL)</p> <p>INVOKE (i)</p> <p>REJECT (j)</p>
DESCRIPTION DU TEST		
1.	Invocation d'une opération liée de Classe 1 du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL initial du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération)
 paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: j (j représente un nombre entier)
 Etiquette d'ID lié: 10000000
 Longueur d'ID lié: 00000001 (un octet)
 ID lié: i
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
 Code d'opération: y (y représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: j
 Etiquette de code de problème: 10000001 (INVOKE)
 Longueur de code de problème: 00000001
 Code de problème: 00000101 (ID lié non reconnu)

NUMÉRO DU TEST: 2.1.3.1.3		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774, 3.7.3/Q.772		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant par la sous-couche composant; Code de problème Return Result		
OBJET: Vérifier qu'une opération simple de Classe 1 peut être invoquée avec succès et que le rejet distant peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke.		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet puisse être créé.		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>ind. TC-INVOKE</i></p> <p><=====</p> <p><i>dem. TC-RESULT-L</i></p> <p>=====></p> <p>RETURN RESULT-LAST (i)</p> <p><i>ind. TC-R-REJECT</i></p> <p><=====</p>	<p>←</p> <p>→</p> <p>←</p>	<p>SP B (CSL)</p> <p>INVOKE (i)</p> <p>REJECT (i)</p>
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 1 du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000000 (ID d'invocation non reconnu)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.3.1.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774, 3.7.4/Q.772		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant par la sous-couche composant; Code de problème Return Error		
OBJET: Vérifier qu'une opération simple de Classe 1 peut être invoquée avec succès et que le rejet distant peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Reject puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>ind. TC-INVOKE</i></p> <p><=====</p> <p><i>dem. TC-U-ERROR</i></p> <p>=====></p> <p>RETURN ERROR (i)</p> <p><i>ind. TC-R-REJECT</i></p> <p><=====</p>	<p>←-----</p> <p>-----></p> <p>←-----</p>	<p>SP B (CSL)</p> <p>INVOKE (i)</p> <p>REJECT (i)</p>
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 1 du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100011 (RETURN ERROR)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'erreur: nombre correct d'octets (par exemple 00000001 si y est long d'un octet)
 Code d'erreur: y
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de code de problème: 10000011 (RETURN ERROR)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (ID d'invocation non reconnu)

NUMÉRO DU TEST: 2.1.3.2.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774, 3.7.2/Q.772		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant par l'utilisateur-TC; Code de problème Invoke		
OBJET: Vérifier qu'un rejet distant par l'utilisateur-TC avec code de problème Invoke peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	_____>	
	<_____	REJECT (i)
<i>ind. TC-U-REJECT</i> <=====		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 1 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans le message de TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération)
paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de code de problème: 10000001 (INVOKE)
Longueur de code de problème: 00000001
Code de problème: 00000000 (ID d'invocation dupliqué)

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de Composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000010 (paramètre type erroné)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.3.2.3		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774, 3.7.4/Q.772		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant par l'utilisateur-TC; Code de problème Return Error		
OBJET: Vérifier qu'un rejet distant par l'utilisateur-TC, avec code de problème Return Error, peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet avec code de problème Return Error puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>ind. TC-INVOKE</i> <=====	←-----	SP B (CSL) INVOKE (i)
<i>dem. TC-U-ERROR</i> =====>	----->	
RETURN ERROR (i) <i>ind. TC-U-REJECT</i> <=====	←-----	REJECT (i)
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 1 du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans le message de TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message de TSL du point SP B au SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100011 (RETURN ERROR)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
 Code d'erreur: y
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000011 (RETURN ERROR)
 Longueur de code de problème: 00000001
 Code de problème: 00000010 (erreur non reconnue)

NUMÉRO DU TEST: 2.1.3.3.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant avec un code de problème Invoke; Invocation d'opération de Classe 1		
OBJET: Vérifier qu'une opération simple de Classe 1 peut être invoquée avec succès et que le rejet distant peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-U-REJECT</i></p> <p>=====</p>	<p>SP B (CSL)</p> <p>—————></p> <p>—————<</p> <p>REJECT (i)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 1 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple., 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération)
paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de code de problème: 10000001 (INVOKE)
Longueur de code de problème: 00000001
Code de problème: 00000010 (paramètre type erroné)

NUMÉRO DU TEST: 2.1.3.3.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant avec un code de problème Invoke; Invocation d'opération de Classe 2		
OBJET: Vérifier qu'une opération simple de Classe 2 peut être invoquée avec succès et que le rejet distant peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	_____>	
	<_____	REJECT (i)
<i>ind. TC-U-REJECT</i> <=====		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 2 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération)
paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de code de problème: 10000001 (INVOKE)
Longueur de code de problème: 00000001
Code de problème: 00000010 (paramètre type erroné)

NUMÉRO DU TEST: 2.1.3.3.3		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant avec un code de problème Invoke; Invocation d'opération de Classe 3		
OBJET: Vérifier qu'une opération simple de Classe 3 peut être invoquée avec succès et que le rejet distant peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	_____>	
	<_____	REJECT (i)
<i>ind. TC-U-REJECT</i> <=====		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 3 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération)
paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de code de problème: 10000001 (INVOKE)
Longueur de code de problème: 00000001
Code de problème: 00000010 (paramètre type erroné)

NUMÉRO DU TEST: 2.1.3.3.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Rejet distant		
SOUS-TITRE: Rejet distant avec un code de problème Invoke; Invocation d'opération de Classe 4		
OBJET: Vérifier qu'une opération simple de Classe 4 peut être invoquée avec succès et que le rejet distant peut être reçu et transmis à l'utilisateur-TC		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Rejet puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-U-REJECT</i></p> <p><=====</p>	<p>SP B (CSL)</p> <p>—————></p> <p><—————</p> <p>REJECT (i)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 4 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération)
paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de code de problème: 10000001 (INVOKE)
Longueur de code de problème: 00000001
Code de problème: 00000010 (paramètre type erroné)

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération non valide)
paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets
Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i
Etiquette de code de problème: 10000001 (type de problème INVOKE)
Longueur de code de problème: 00000001
Code de problème: 00000001 (opération non reconnue)

NUMÉRO DU TEST: 2.1.4.1.2		Feuillet: 1 de 3
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Fonctions valides; Réception d'un composant entraînant un rejet par l'utilisateur-TC		
SOUS-TITRE: Problème Invoke; Opération liée non prévue		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause d'une opération liée non prévue, sans que la demande d'invocation d'opération initiale en soit affectée		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Invoke avec un ID lié soit contenu dans un message TSL approprié		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————>	
<i>ind. TC-INVOKE</i> <=====	<—————	INVOKE (j, i)
<i>dem. TC-U-REJECT</i> =====>		
REJECT (j)	—————>	
<i>ind. TC-RESULT-L</i> <=====	<—————	RETURN RESULT-LAST (i)
DESCRIPTION DU TEST		
1.	Invocation d'une opération non liée du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT INVOKE LIÉ A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
5.	VÉRIFICATION D: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT REJECT ÉTAIT-IL IDENTIQUE À L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT INVOKE ENVOYÉ PAR LE POINT SP B?	
6.	VÉRIFICATION E: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyé par le point SP B

Etiquette de type composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: j (j représente un entier)

Etiquette d'ID lié: 10000000

Longueur d'ID lié: 00000001 (un octet)

ID lié: i (i est un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: y (y représente un code d'opération non lié à x)

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL envoyé par le point SP A

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j

Etiquette de code de problème: 10000001 (INVOKE)

Longueur de code de problème: 00000001

Code de problème: 00000111 (opération liée non prévue)

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

Paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.4.1.3		Feuillet: 1 de 3
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Réception d'un composant entraînant un rejet par l'utilisateur-TC		
SOUS-TITRE: Problème Invoke; Réponse liée non prévue		
OBJET: Vérifier qu'une réponse liée non prévue peut être rejetée		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke qui invoquera une opération liée</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'une réponse liée contienne au moins un paramètre non associé au résultat de l'opération</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-INVOKE</i></p> <p><=====</p> <p><i>dem. TC-U-REJECT</i></p> <p>=====></p> <p>REJECT (j)</p> <p><i>ind. TC-RESULT-L</i></p> <p><=====</p>	<p>SP B (CSL)</p> <p>INVOKE (j, i)</p> <p>RETURN RESULT-LAST (i)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT INVOKE LIÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
5.	VÉRIFICATION D: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
6.	VÉRIFICATION E: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide qui ne permet aucune opération liée)

paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyé du point SP B au point SP A

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: j (j représente un nombre entier)

Etiquette d'ID lié: 10000000

Longueur d'ID lié: 00000001 (un octet)

ID lié: i

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'opération: y (y représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL envoyé par le point SP A

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code de problème: 10000001 (INVOKE)
Longueur de code de problème: 00000001
Code de problème: 00000111 (réponse liée non prévue)

Composant RETURN RESULT-LAST dans le message TSL envoyé par le point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)
Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
(voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC, y compris au moins un paramètre non associé à l'opération)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets
Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i
Etiquette de code de problème: 10000001 (type de problème Invoke)
Longueur de code de problème: 00000001
Code de problème: 00000010 (paramètre type erroné)

NUMÉRO DU TEST: 2.1.4.2.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Fonctions valides; Réception d'un composant entraînant un rejet par l'utilisateur-TC		
SOUS-TITRE: Problème Return Result; Paramètre type erroné		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause d'un code d'opération non valide inclus dans le composant Return Result-Last		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke pour la Classe 1 ou 3 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Last, avec un code d'opération non valide, soit créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————>	
	<—————	RETURN RESULT-LAST (i)
<i>ind. TC-RESULT-L</i> <=====		
<i>dem. TC-U-REJECT</i> =====>		
REJECT (i)	—————>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP A au point SP B. Créer une réponse du point SP B au point SP A avec un Identificateur Invoke ID valide mais avec un code d'opération différent.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT-LAST A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
5.	VÉRIFICATION D: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet) (voir la Note)

Code d'opération: y (y est différent de x) (voir la Note)

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code de problème: 10000010 (RETURN RESULT)

Longueur de code de problème: 00000001

Code de problème: 00000010 (paramètre type erroné)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.4.3.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Fonctions valides; Réception d'un composant entraînant un rejet par l'utilisateur-TC		
SOUS-TITRE: Problème Return Error; Erreur non reconnue		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause d'un code d'erreur non reconnu inclus dans le composant Return Error		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke pour la Classe 1</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error avec un code d'erreur non valide soit créé</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-U-ERROR</i></p> <p><=====</p> <p><i>dem. TC-U-REJECT</i></p> <p>=====></p> <p>REJECT (i)</p>	<p>SP B (CSL)</p> <p>RETURN ERROR (i)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 1 du point SP A au point SP B. Créer une réponse négative du point SP B au point SP A avec un Identificateur Invoke ID valide mais avec un code d'erreur non valide pour cette opération.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN ERROR A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
5.	VÉRIFICATION D: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL envoyé du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'erreur: y (y est un code d'erreur non valide pour cette opération)

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL envoyé du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code de problème: 10000011 (RETURN ERROR)

Longueur de code de problème: 00000001

Code de problème: 00000010 (erreur non reconnue)

NUMÉRO DU TEST: 2.1.4.3.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Fonctions valides; Réception d'un composant entraînant un rejet par l'utilisateur-TC		
SOUS-TITRE: Problème Return Error; Erreur non prévue		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause d'un code d'erreur non prévu inclus dans le composant Return Error		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke pour la Classe 1</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error avec un code d'erreur imprévu soit créé</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-U-ERROR</i></p> <p><=====</p> <p><i>dem. TC-U-REJECT</i></p> <p>=====></p> <p>REJECT (i)</p>	<p>SP B (CSL)</p> <p>RETURN ERROR (i)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 1 du point SP A au point SP B. Créer une réponse négative du point SP B au point SP A avec un Identificateur Invoke ID valide mais avec un code d'erreur non prévu pour cette opération.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN ERROR A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
5.	VÉRIFICATION D: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL envoyé du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation : 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'erreur: y (y est un code d'erreur non inclus dans les codes d'erreur qui peuvent être signalés au titre de l'opération invoquée)

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code de problème: 10000011 (RETURN ERROR)

Longueur de code de problème: 00000001

Code de problème: 00000011 (erreur non prévue)

NUMÉRO DU TEST: 2.1.4.3.3		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Fonctions valides; Réception d'un composant entraînant un rejet par l'utilisateur-TC		
SOUS-TITRE: Problème Return Error; Paramètre type erroné		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause d'un paramètre type erroné inclus dans le composant Return Error		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke de Classe 1 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error avec un paramètre type erroné soit créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====> INVOKE (i)	-----> -----<	SP B (CSL) RETURN ERROR (i)
<i>ind. TC-U-ERROR</i> <=====		
<i>dem. TC-U-REJECT</i> =====> REJECT (i)	----->	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 1 du point SP A au point SP B. Créer une réponse négative du point SP B au point SP A avec un Identificateur Invoke ID valide mais avec un paramètre type erroné pour cette opération.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN ERROR A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
5.	VÉRIFICATION D: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'erreur: y (y est un code d'erreur valide pour cette opération)

paramètres (fournis par l'utilisateur-TC, y compris au moins une Etiquette de paramètre non incluse dans les Etiquettes associées au résultat de l'opération)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code de problème: 10000011 (RETURN ERROR)

Longueur de code de problème: 00000001

Code de problème: 00000100 (paramètre type erroné)

NUMÉRO DU TEST: 2.1.5.1.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Segmentation du composant Return Result		
SOUS-TITRE: Invocation d'opération simple de Classe 1; IUT émetteur; réception des composants segmentés		
OBJET: Vérifier qu'une opération simple de Classe 1 peut être exécutée à la suite de la réception de composants Return Result segmentés		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result Not-last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————→	
	←—————	RETURN RESULT NOT- LAST (i)
<i>ind. TC-RESULT-NL</i> <=====		
	←—————	RETURN RESULT-LAST (i)
<i>ind. TC-RESULT-L</i> <=====		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT NOT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT NOT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.5.1.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Segmentation du composant Return Result		
SOUS-TITRE: Invocation d'opération simple de Classe 1; IUT récepteur; émission des composants segmentés		
OBJET: Vérifier qu'une opération simple de Classe 1 peut être exécutée par l'envoi de composants Return Result segmentés		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke 2) Provoquer une stimulation de l'utilisateur-TC telle qu'un composant Return Result Not-last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>ind. TC-INVOKE</i> <===== <i>dem. TC-RESULT-NL</i> =====> RETURN RESULT NOT-LAST (i) <i>dem. TC-RESULT-L</i> =====> RETURN RESULT LAST (i)	← → →	SP B (CSL) INVOKE (i)
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT NOT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT NOT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 Paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.5.2.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Segmentation du composant Return Result		
SOUS-TITRE: Invocation d'opération simple de Classe 3; IUT émetteur: réception des composants segmentés		
OBJET: Vérifier qu'une opération simple de Classe 3 peut être exécutée à la suite de la réception de composants Return Result segmentés		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result Not-Last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————→	
	←—————	RETURN RESULT NOT-LAST (i)
<i>ind. TC-RESULT-NL</i> <=====		
	←—————	RETURN RESULT-LAST (i)
<i>ind. TC-RESULT-L</i> <=====		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple de Classe 3 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT NOT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT NOT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.6		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides		
SOUS-TITRE: Annulation par l'utilisateur		
OBJET: Vérifier qu'une invocation d'opération peut être annulée par l'utilisateur-TC		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Last puisse être créé</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) —————></p> <p><i>dem. TC-U-CANCEL</i></p> <p>=====></p> <p><————— RETURN-RESULT-LAST (i)</p> <p><i>ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (i) —————></p>		
DESCRIPTION DU TEST		
<ol style="list-style-type: none"> 1. Invocation d'une opération simple de Classe 1 du point SP A au point SP B. Faire en sorte que l'utilisateur-TC annule l'opération immédiatement après l'envoi du composant Invoke. 2. VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A? 3. VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A? 4. VÉRIFICATION C: LE FLUX DE COMPOSANTS ÉTAIT-IL COMME INDIQUÉ CI-DESSUS? 5. VÉRIFICATION D: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A? 		
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets Composant INVOKE dans le message TSL du point SP A au point SP B Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000001 (un octet) ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000000 (ID d'invocation non reconnu)
 NOTE – Omis lorsque aucun paramètre n'est présent.

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100011 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.7.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.3/Q.773		
TITRE: Fonctions valides; Variations de codage		
SOUS-TITRE: Longueur de composant: longue définie		
OBJET: Vérifier qu'une partie composant avec une forme longue définie peut être acceptée		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke		
2) Faire en sorte que les données au point SP A soient telles qu'un composant Return Result-Last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL)		SP B (CSL)
	←	INVOKE (i)
<i>ind. TC-INVOKE</i>		
←=====		
<i>dem. TC-RESULT-L</i>		
=====→		
RETURN RESULT-LAST (i)	→	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 1 ou 3 du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets (forme longue définie)		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
Longueur de composant: nombre correct d'octets
Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i
Etiquette de séquence: 00000010 (voir la Note)
Longueur de séquence: nombre correct d'octets (voir la Note)
Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
Code d'opération: x (voir la Note)
paramètres (fournis par l'utilisateur-TC)
NOTE – Omis lorsque aucun paramètre n'est présent.

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)
 Etiquette EOC: 00000000
 Longueur EOC: 00000000

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.7.4.1.1		Feuillet: 1 de 2
RÉFÉRENCE: 6.2/Q.773		
TITRE: Fonctions valides; Variations de codage		
SOUS-TITRE: Variations de valeur; Invoke ID; Invoke ID = -127 (FFh)		
OBJET: Vérifier que l'IUT (point SP A) peut traiter un codage correct d'un ID de composant (valeur supérieure)		
CONDITIONS INITIALES: Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ← ind. TC-INVOKE <===== dem. TC-RESULT-L =====→ RETURN RESULT-LAST (i) → </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP B au point SP A avec identificateur Invoke ID positionné à 11111111.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
5.	VÉRIFICATION D: LE CODE D'OPÉRATION DANS LE COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: 11111111 (FFh)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: 11111111 (FFh)
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.7.4.1.2		Feuillet: 1 de 2
RÉFÉRENCE: 6.2/Q.773		
TITRE: Fonctions valides; Variations de codage		
SOUS-TITRE: Variations de valeur; Invoke ID; Invoke ID = 0 (00h)		
OBJET: Vérifier que l'IUT (point SP A) peut traiter un codage correct d'un ID de composant (valeur inférieure)		
CONDITIONS INITIALES: Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ← INVOKE (i) ind. TC-INVOKE <===== dem. TC-RESULT-L =====> RETURN RESULT-LAST (i) → </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP B au point SP A avec identificateur Invoke ID positionné à 0.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT RETURN RESULT-LAST AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
5.	VÉRIFICATION D: LE CODE D'OPÉRATION DANS LE COMPOSANT RETURN RESULT-LAST ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001		
ID d'invocation: 0		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: 0
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.7.4.2		Feuillet: 1 de 2
RÉFÉRENCE: 6.3/Q.773		
TITRE: Fonctions valides; Variations de codage		
SOUS-TITRE: Variations de valeur; Code d'opération global		
OBJET: Vérifier qu'un code d'opération global peut être décodé correctement par le sous-système TCAP		
CONDITIONS INITIALES: Provoquer une stimulation telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke avec un code d'opération global. La valeur globale ne correspond pas à une opération prise en charge		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>ind. TC-INVOKE</i></p> <p><=====</p> <p><i>dem. TC-U-REJECT</i></p> <p>=====></p> <p>REJECT (i)</p>	<p>←</p> <p>→</p>	<p>SP B (CSL)</p> <p>INVOKE (i)</p>
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP B au point SP A avec un code d'opération global non pris en charge.	
2.	VÉRIFICATION A: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT REJECT ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP B au point SP A		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		
Etiquette de code d'opération: 00000110 (globale)		
Longueur de code d'opération: 00000011 (3)		
Code d'opération: 0000 0000		
0001 0001		
1000 0101		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code de problème: 10000001 (Type de problème INVOKE)

Longueur de code de problème: 00000001

Code de problème: 00000001 (opération non reconnue)

NUMÉRO DU TEST: 2.1.8.1		Feuillet: 1 de 2
RÉFÉRENCE: Q.774		
TITRE: Fonctions valides; Groupement de composants multiples		
SOUS-TITRE: Invocation d'opérations multiples: réception succès		
OBJET: Vérifier que des opérations multiples peuvent être invoquées avec succès et que les indications positives d'exécution de ces opérations peuvent être reçues correctement		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne des composants multiples 2) Faire en sorte que l'utilisateur-TC au point SP B envoie des indications positives d'exécution des opérations avec un message TSL approprié		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TC-INVOKE (#1)</i></p> <p>=====></p> <ul style="list-style-type: none"> • • • <p><i>dem. TC-INVOKE (#n)</i></p> <p>=====></p> <p style="text-align: center;">INVOKE #1, ..., #n^{a)}</p> <p style="text-align: center;">-----></p> <p style="text-align: center;">RETURN RESULT-LAST #1, ..., #n^{a)}</p> <p style="text-align: center;">-----<</p> <p><i>ind. TC-RESULT-L (#1)</i></p> <p><=====</p> <ul style="list-style-type: none"> • • • <p><i>ind. TC-RESULT-L (#n)</i></p> <p>=====></p>		
^{a)} La séquence des composants est déterminée par l'utilisateur-TC. NOTE – Le nombre de composants dépend de l'utilisateur-TC.		
DESCRIPTION DU TEST		
1.	Invocation d'opérations multiples dans un message TSL du point SP A au point SP B.	
2.	VÉRIFICATION A: TOUS LES COMPOSANTS INVOKE DANS UN MESSAGE TSL ONT-ILS ÉTÉ ENVOYÉS AVEC INFORMATION CORRECTE PAR LE POINT SP A?	
3.	VÉRIFICATION B: TOUS LES COMPOSANTS RETURN RESULT-LAST DANS UN MESSAGE TSL ONT-ILS ÉTÉ TRANSMIS À L'UTILISATEUR-TC DANS L'ORDRE OÙ ILS ONT ÉTÉ FOURNIS, AVEC INFORMATION CORRECTE, PAR LE POINT SP B?	
4.	VÉRIFICATION C: TOUS LES AUTOMATES À ÉTATS D'INVOCATION (1, ..., n) ÉTAIENT-ILS AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: 1, ou, ..., n correspondant aux composants INVOKE #1, ..., #n

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x1, ..., xn représentant des codes d'opération valides

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: 1, ou, ..., n

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x1, ou, ..., xn (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.8.2	Feuillet: 1 de 2	
RÉFÉRENCE: Q.774		
TITRE: Fonctions valides; Groupement de composants multiples		
SOUS-TITRE: Invocation d'opérations multiples: notification succès		
OBJET: Vérifier que des opérations multiples peuvent être invoquées avec succès et que les indications positives d'exécution de ces opérations peuvent être envoyées		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne des composants multiples 2) Faire en sorte que l'utilisateur-TC au point SP A envoie des indications positives d'exécution des opérations avec un message TSL approprié		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL)</p> <p style="text-align: center;">INVOKE #1, ..., #n^{a)}</p> <p style="text-align: center;">←—————</p> <p><i>ind. TC-INVOKE (#1)</i></p> <p>←=====</p> <ul style="list-style-type: none"> • • • <p><i>ind. TC-INVOKE (#n)</i></p> <p>←=====</p> <p><i>dem. TC-RESULT-L (#1)</i></p> <p>=====></p> <ul style="list-style-type: none"> • • • <p><i>dem. TC-RESULT-L (#n)</i></p> <p>=====></p> <p style="text-align: center;">RETURN RESULT-LAST #n, ..., #1^{a)}</p> <p style="text-align: center;">—————→</p>		
<p>a) La séquence des composants est déterminée par l'utilisateur-TC.</p> <p>NOTE – Le nombre de composants dépend de l'utilisateur-TC.</p>		
DESCRIPTION DU TEST		
1.	Invocation d'opérations multiples dans un message TSL du point SP B au point SP A.	
2.	VÉRIFICATION A: TOUS LES COMPOSANTS INVOKE DANS UN MESSAGE TSL ONT-ILS ÉTÉ TRANSMIS À L'UTILISATEUR-TC DANS L'ORDRE OÙ ILS ONT ÉTÉ FOURNIS, AVEC INFORMATION CORRECTE, PAR LE POINT SP B?	
3.	VÉRIFICATION B: TOUS LES COMPOSANTS RETURN RESULT-LAST DANS UN MESSAGE TSL ONT-ILS ÉTÉ ENVOYÉS AVEC INFORMATION CORRECTE PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'IDENTIFICATEUR INVOKE ID DANS CHACUN DES COMPOSANTS RETURN RESULT-LAST CORRESPONDAIT-IL D'UNE MANIÈRE BIUNIVOQUE À CELUI DE CHACUN DES COMPOSANTS INVOKE?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: 1, ou, ..., n correspondant aux composants INVOKE #1, ..., #n

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x1, ou, ..., xn représentant des codes d'opération valides

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: 1, ou, ..., n

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x1, ou, ..., xn (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.1.8.3		Feuillet: 1 de 3
RÉFÉRENCE: 3.2.2.2/Q.774		
TITRE: Fonctions valides; Groupement de composants multiples		
SOUS-TITRE: Composant mal formé reçu		
OBJET: Vérifier que des composants ultérieurs dans le message peuvent être ignorés lorsqu'un composant mal structuré est détecté par la sous-couche composant		
CONDITIONS INITIALES: Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne des composants multiples dont le deuxième est mal structuré		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <p>SP A (CSL) SP B (CSL)</p> <p style="text-align: center;">INVOKE #1, #2, #3 (Note 1)</p> <p style="text-align: center;">←</p> <p style="text-align: center;">(#2 mal structuré, par exemple code d'opération manquant)</p> <p><i>ind. TC-INVOKE (#1)</i></p> <p>=====</p> <p><i>ind. TC-L-REJECT (#2)</i></p> <p>=====</p> <p><i>dem. TC-RESULT-L (#1)</i></p> <p>=====→</p> <p style="text-align: center;">REJECT #2, RETURN RESULT-LAST #1 (Note 2)</p> <p style="text-align: center;">→</p> <p>NOTE 1 – La séquence des composants Invoke est importante. NOTE 2 – La séquence de ces composants n'est pas importante.</p>		
DESCRIPTION DU TEST		
1.	Invocation d'opérations multiples dans un message TSL du point SP B au point SP A dans l'ordre indiqué par le diagramme.	
2.	VÉRIFICATION A: LE PREMIER COMPOSANT INVOKE A-T-IL ÉTÉ TRANSMIS À L'UTILISATEUR-TC?	
3.	VÉRIFICATION B: SEULS LE COMPOSANT RETURN RESULT-LAST POUR LA PREMIÈRE OPÉRATION ET LE COMPOSANT REJECT POUR LA DEUXIÈME OPÉRATION ONT-ILS ÉTÉ ENVOYÉS PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Composants INVOKE #1 dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100001 (INVOKE)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: 1
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x est un code d'opération valide
 paramètres (fournis par l'utilisateur-TC)

Composant INVOKE #2 dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100001 (INVOKE)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: 2
 paramètres (fournis par l'utilisateur-TC)

Composant INVOKE #3 dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100001 (INVOKE)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: 3
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x est un code d'opération valide
 paramètres (fournis par l'utilisateur-TC)

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Composant RETURN RESULT-LAST #1 dans un message TSL envoyé du point SP A au point SP B

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'Identificateur d'invocation: 00000010

Longueur d'Identificateur d'invocation: 00000001

Identificateur d'invocation: 1

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (local) ou 00000110 (global) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est d'une longueur d'un octet) (voir la Note)

Code d'opération: x

paramètres (fournis par l'utilisateur du TC)

Composant REJECT #2 dans le message TSL envoyé du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'Identificateur d'invocation: 00000010

Longueur d'Identificateur d'invocation: 00000001

Identificateur d'invocation: 2

Etiquette de code de problème: 10000000 (problème général)

Longueur de code de problème: 00000001

Code de problème: 00000010 (composant mal formé)

NOTE – Omis lorsqu'aucun paramètre n'est présent..

NUMÉRO DU TEST: 2.1.9.1.1		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Accepter la proposition de contexte d'application; Envoyer AARQ dans le message Begin		
OBJET: Vérifier qu'une IUT peut générer et envoyer l'unité APDU AARQ de commande de dialogue dans la partie dialogue dans un message Begin		
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN (AARQ) —————></p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie Dialogue.	
2.	<p>VÉRIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE BEGIN CONTIENT-ELLE L'UNITE APDU AARQ?</p> <p>Faire également en sorte qu'un message END soit envoyé par l'IUT ou par le Testeur. Lorsque le dernier message a été reçu par le Testeur, le message END envoyé par l'IUT peut être utilisé pour vérifier que le dernier message a bien été reçu.</p>	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE		
<p>Partie dialogue dans le message Begin</p> <p>Etiquette partie dialogue: 01101011</p> <p>Longueur de partie dialogue: nombre correct d'octets</p> <p>Type de données externes dans la partie dialogue</p> <p>Etiquette de type externe: 00101000</p> <p>Longueur externe: nombre correct d'octets</p> <p>Etiquette d'identificateur d'objet: 00000110</p> <p>Longueur d'identificateur d'objet: 00000111</p> <p>Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré)</p> <p>Etiquette de type ASN.1 individuel: 10100000</p> <p>Longueur de type ASN.1 individuel: nombre correct d'octets</p> <p>Dialogue PDU</p> <p>Etiquette de demande de dialogue: 01100000</p> <p>Longueur de demande de dialogue: nombre correct d'octets</p> <p>Etiquette de contexte d'application: 10100001</p> <p>Longueur de contexte d'application: nombre correct d'octets</p> <p>Etiquette d'identificateur d'objet: 00000110</p> <p>Longueur d'identificateur d'objet: nombre correct d'octets</p> <p>Référence directe: tout identificateur d'objet</p>		

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Begin

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000 (voir la Note)

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Demande de Dialogue: 01100000

Longueur de la Demande de Dialogue: nombre correct d'octets

Etiquette de Version du Protocole: 10000000

Longueur de Version du Protocole: 00000010

Valeur de Version du Protocole: 00000111 10000000

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: tout identificateur d'objet

NOTE – Au lieu de l'option de codage de type ASN.1 individuel, ce test peut également être réalisé avec les deux autres options – octet aligné et arbitraire.

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Begin

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: mêmes octets que dans AARQ

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000000 (Accepté)

Etiquette de diagnostic d'origine du résultat: 10100011

Longueur de diagnostic de l'origine du résultat: 00000101

Etiquette de l'utilisateur du service de dialogue: 10100001

Longueur de l'utilisateur du service de dialogue: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur de l'utilisateur du service de dialogue: 00000000 (Néant)

NUMÉRO DU TEST: 2.1.9.1.3		Feuillet: 1 de 2			
RÉFÉRENCE: 3.2.1.2/Q.774					
TITRE: Fonctions valides; Partie Dialogue					
SOUS-TITRE: Accepter la proposition de contexte d'application; Accepter AARQ et terminer le dialogue					
OBJET: Vérifier qu'une IUT peut recevoir un message Begin avec l'unité APDU AARQ, puis générer et envoyer l'unité APDU 'AARE' dans la partie dialogue d'un message End.					
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos					
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; vertical-align: top;"> <p>SP A (CSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p>END (AARE)</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p>←-----</p> <p>-----→</p> </td> <td style="width: 40%; vertical-align: top;"> <p>SP B (CSL)</p> <p>BEGIN (AARQ)</p> </td> </tr> </table>			<p>SP A (CSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p>END (AARE)</p>	<p>←-----</p> <p>-----→</p>	<p>SP B (CSL)</p> <p>BEGIN (AARQ)</p>
<p>SP A (CSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>←=====</p> <p><i>dem. TR-END</i></p> <p>=====→</p> <p>END (AARE)</p>	<p>←-----</p> <p>-----→</p>	<p>SP B (CSL)</p> <p>BEGIN (AARQ)</p>			
DESCRIPTION DU TEST					
1.	Faire en sorte que le point SP B envoie un message Begin comportant une demande de dialogue				
2.	VÉRIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE END CONTIENT-ELLE L'UNITÉ APDU AARE ET LE CONTEXTE D'APPLICATION EST-IL LE MEME QUE DANS L'AARQ REÇU?				

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Begin

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré)

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: mêmes octets que dans AARQ

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000000 (Accepté)

Etiquette de diagnostic d'origine du résultat: 10100011

Longueur de diagnostic de l'origine du résultat: 00000101

Etiquette de l'utilisateur du service de dialogue: 10100001

Longueur de l'utilisateur du service de dialogue: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur de l'utilisateur du service de dialogue: 00000000 (Néant)

NUMÉRO DU TEST: 2.1.9.2.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Proposer un contexte d'application différent; Envoyer APDU AARE avec une proposition alternative		
OBJET: Vérifier qu'une IUT peut générer et envoyer une unité APDU 'AARE" de commande de dialogue avec un contexte d'application différent dans la partie dialogue d'un message Continue		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
	←	BEGIN (AARQ)
<i>ind. TR-BEGIN</i>		
←		
<i>dem. TR-CONTINUE</i>		
→		
CONTINUE (AARE)	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une demande de dialogue.	
2.	Faire en sorte que le point SP A propose un contexte d'application différent.	
3.	VÉRIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE CONTINUE CONTIENT-ELLE L'UNITE APDU AARE ET LE CONTEXTE D'APPLICATION EST-IL DIFFERENT DE CELUI CONTENU DANS L'AARQ REÇU?	
	Faire également en sorte qu'un message END soit envoyé par l'IUT ou par le Testeur. Lorsque le dernier message a été envoyé par le Testeur, le message END envoyé par l'IUT peut être utilisé pour vérifier que le dernier message a bien été reçu.	

TABLE DE VERIFICATION DES ELEMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: H'00118605010101 (syntaxe abstraite de dialogue structuré)

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: nombre correct d'octets différents de ceux dans l'AARQ

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000000 (Accepté)

Etiquette de diagnostic d'origine du résultat: 10100011

Longueur de diagnostic de l'origine du résultat: 00000101

Etiquette de l'utilisateur du service de dialogue: 10100001

Longueur de l'utilisateur du service de dialogue: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur de l'utilisateur du service de dialogue: 00000000 (Néant)

NUMÉRO DU TEST: 2.1.9.2.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Proposer un contexte d'application; Recevoir APDU AARE avec la proposition alternative		
OBJET: Vérifier qu'une IUT peut accepter une unité APDU 'AARE' de commande de dialogue avec un contexte d'application différent dans la partie dialogue d'un message Continue.		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>dem. TR-BEGIN</i>		
=====>		
BEGIN (AARQ)	—————>	
	←—————	CONTINUE (AARE)
<i>ind. TR-CONTINUE</i>		
=====<		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue proposant un contexte d'application différent.	
3.	VÉRIFICATION A: L'IUT ACCEPTE-T-ELLE L'UNITE APDU 'AARE' AVEC UN CONTEXTE D'APPLICATION DIFFERENT? Faire également en sorte qu'un message END soit envoyé par l'IUT ou par le Testeur. Lorsque le dernier message a été envoyé par le Testeur, le message END envoyé par l'IUT peut être utilisé pour vérifier que le dernier message a bien été reçu.	

TABLE DE VERIFICATION DES ELEMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011
 Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000
 Longueur externe: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: 00000111
 Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré)

Etiquette de type ASN.1 individuel: 10100000
 Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001
 Longueur de Réponse de Dialogue: nombre correct d'octets
 Etiquette de contexte d'application: 10100001
 Longueur de contexte d'application: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: nombre correct d'octets
 Référence directe: nombre correct d'octets différents de ceux dans AARQ

Etiquette Résultat: 10100010
 Longueur Résultat: 00000011
 Etiquette de type entier: 00000010
 Longueur d'entier: 00000001
 Valeur Résultat: 00000000 (Accepté)

Etiquette de diagnostic d'origine du résultat: 10100011
 Diagnostic de l'origine du résultat Longueur: 00000101
 Etiquette de l'utilisateur du service de dialogue: 10100001
 Longueur de l'utilisateur du service de dialogue: 00000011
 Etiquette de type entier: 00000010
 Longueur d'entier: 00000001
 Valeur de l'utilisateur du service de dialogue: 00000000 (Néant)

NUMÉRO DU TEST: 2.1.9.3		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Dialogue refusé		
OBJET: Vérifier qu'une IUT peut générer et envoyer une unité APDU 'AARE' de commande de dialogue dans la partie dialogue d'un message Abort pour indiquer que le contexte d'application n'est pas pris en charge		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
	←	BEGIN (AARQ)
<i>ind. TR-BEGIN</i>		
←		
<i>dem. TR-U-ABORT</i>		
=====>		
ABORT (AARE)	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une demande de dialogue.	
2.	Faire en sorte que le point SP A refuse le dialogue du fait que le contexte d'application n'est pas pris en charge.	
3.	VÉRIFICATION A: LE POINT SP A A-T-IL TRANSMIS LE MESSAGE ABORT ATTENDU AVEC L'UNITE APDU 'AARE'?	

TABLE DE VERIFICATION DES ELEMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Abort

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré)

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

référence directe: mêmes octets que dans la demande de dialogue

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000001 (Rejet – Permanent)

Etiquette de diagnostic d'origine du résultat: 10100011

Longueur d'origine du résultat: 00000101

Etiquette de l'utilisateur du service de dialogue: 10100001

Longueur de l'utilisateur du service de dialogue: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur de l'utilisateur du service de dialogue: 00000010 (Contexte d'application pas pris en charge)

NUMÉRO DU TEST: 2.1.9.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Dialogue abandonné		
OBJET: Vérifier qu'une IUT peut générer et envoyer l'unité APDU 'ABRT' de commande de dialogue dans la partie dialogue d'un message Abort après établissement du dialogue		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN (AARQ) —————></p> <p>—————< CONTINUE (AARE)</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====<</p> <p><i>dem. TR-U-ABORT</i></p> <p>=====></p> <p>ABORT (U) (ABRT) —————></p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue.	
3.	Faire en sorte que le point SP A abandonne le dialogue pour une raison quelconque.	
4.	VERIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE ABORT CONTIENT-ELLE L'UNITE APDU 'ABRT'?	

TABLE DE VERIFICATION DES ELEMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Abort.

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré)

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de dialogue abandonné: 01100100

Longueur de dialogue abandonné: 00000011

Etiquette d'origine de l'abandon: 10000000

Longueur de l'origine de l'abandon: 00000001

Origine Abandon: 00000000 (utilisateur du service de dialogue)

NUMÉRO DU TEST: 2.1.9.5.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Transport de l'information d'utilisateur; Accepter l'information d'utilisateur dans un message Begin		
OBJET: Vérifier qu'une IUT peut recevoir un message Begin avec l'unité APDU AARQ comportant un élément d'Information d'utilisateur		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p style="text-align: center;">←</p> <p><i>ind. TR-BEGIN</i> BEGIN (AARQ)</p> <p>←=====</p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin avec une demande de dialogue comportant une Information d'utilisateur.	
2.	<p>VÉRIFICATION A: L'IUT ACCEPTE-T-ELLE L'INFORMATION D'UTILISATEUR DANS LA DEMANDE DE DIALOGUE REÇUE?</p> <p>Faire également en sorte qu'un message END soit envoyé par l'IUT. Le message END envoyé par l'IUT peut être utilisé pour vérifier que le dernier message a bien été reçu.</p>	

TABLE DE VERIFICATION DES ELEMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie de dialogue dans le message Begin

Etiquette partie dialogue: 01101011
 Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000
 Longueur externe: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: 00000111
 Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré)

Etiquette de type ASN.1 individuel: 10100000
 Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Demande de Dialogue: 01100000
 Longueur de Demande de Dialogue: nombre correct d'octets
 Etiquette de contexte d'application: 10100001
 Longueur de contexte d'application: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: nombre correct d'octets
 Référence directe: tout identificateur d'objet

Information d'utilisateur dans le dialogue PDU

Etiquette d'information d'utilisateur: 10111110
 Longueur d'information d'utilisateur: nombre correct d'octets
 Etiquette de type externe: 00101000
 Longueur externe: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: nombre correct d'octets
 Référence directe: tout identificateur d'objet

Etiquette de type ASN.1 individuel: 10100000 (voir la Note)
 Longueur de type ASN.1 individuel: nombre correct d'octets

Certains octets de données utilisateur dans la syntaxe abstraite définie par l'utilisateur

NOTE – Ce test peut également être réalisé avec les 2 autres options de codage-octet aligné et arbitraire.

NUMÉRO DU TEST: 2.1.9.5.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Transport de l'information d'utilisateur; Accepter l'information d'utilisateur dans le premier message Continue		
OBJET: Vérifier qu'une IUT peut accepter un message Continue avec l'unité APDU 'AARE' comportant un élément d'information d'utilisateur		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (CSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN (AARQ)</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p>	<p>SP B (CSL)</p> <p>—————></p> <p>—————<</p> <p>CONTINUE (AARE)</p>	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue comportant une information d'utilisateur dans la partie dialogue.	
3.	<p>VÉRIFICATION A: L'IUT ACCEPTE-T-ELLE L'INFORMATION D'UTILISATEUR DANS LA REPONSE DE DIALOGUE REÇUE?</p> <p>Faire également en sorte qu'un message END soit envoyé par l'IUT ou par le Testeur. Lorsque le dernier message a été envoyé par le testeur, le message END envoyé par l'IUT peut être utilisé pour vérifier que le dernier message a bien été reçu.</p>	

TABLE DE VERIFICATION DES ELEMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011
 Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000
 Longueur externe: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: 00000111
 Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré)

Etiquette de type ASN.1 individuel: 10100000
 Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001
 Longueur de Réponse de Dialogue: nombre correct d'octets
 Etiquette de contexte d'application: 10100001
 Longueur de contexte d'application: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: nombre correct d'octets
 Référence directe: tout identificateur d'objet
 Etiquette Résultat: 10100010
 Longueur Résultat: 00000011
 Etiquette de type entier: 00000010
 Longueur d'entier: 00000001
 Valeur Résultat: 00000000 (Accepté)
 Etiquette de diagnostic d'origine du résultat: 10100011
 Longueur de diagnostic de l'origine du résultat: 00000101
 Etiquette de l'utilisateur du service de dialogue: 10100001
 Longueur de l'utilisateur du service de dialogue: 00000011
 Etiquette de type entier: 00000010
 Longueur d'entier: 00000001
 Valeur de l'utilisateur du service de dialogue: 00000000 (NUL)

Information d'utilisateur dans le dialogue PDU

Etiquette d'information d'utilisateur: 10111110
 Longueur d'information d'utilisateur: nombre correct d'octets
 Etiquette de type externe: 00101000
 Longueur externe: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: nombre correct d'octets
 Référence directe: tout identificateur d'objet
 Etiquette de type ASN.1 individuel: 10100000
 Longueur de type ASN.1 individuel: nombre correct d'octets
 Certains octets de données utilisateur dans la syntaxe abstraite définie par l'utilisateur

NUMÉRO DU TEST: 2.1.9.5.3		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Transport de l'information d'utilisateur; Accepter l'information d'utilisateur dans un message Continue ultérieur		
OBJET: Vérifier que le point SP A peut accepter un élément d'information d'utilisateur dans un message Continue ultérieur		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos et les cas de test 2.1.9.5.2 doivent être réussis		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
	←	BEGIN (AARQ)
<i>ind. TR-BEGIN</i>		
←=====		
<i>dem. TR-CONTINUE</i>		
=====→		
CONTINUE (AARE)	→	
	←	CONTINUE
<i>ind. TR-CONTINUE</i>		
←=====		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin avec la demande de dialogue au point SP A.	
2.	Faire en sorte que le point SP A confirme the dialogue.	
3.	Faire en sorte que le point SP B envoie un message Continue comportant un élément d'information d'utilisateur en tant que partie dialogue.	
4.	VÉRIFICATION A: VERIFIER QUE L'IUT AU POINT SP A A ACCEPTE L'ELEMENT D'INFORMATION D'UTILISATEUR Faire également en sorte qu'un message END soit envoyé par l'IUT ou par le Testeur. Lorsque le dernier message a été envoyé par le Testeur, le message END envoyé par l'IUT peut être utilisé pour vérifier que le dernier message a bien été reçu.	
TABLE DE VERIFICATION DES ELEMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE		
Partie dialogue dans le message Continue		
Etiquette partie dialogue: 01101011		
Longueur de partie dialogue: nombre correct d'octets		
Type de données externes dans la partie dialogue (élément d'information d'utilisateur)		
Etiquette de type externe: 00101000		
Longueur externe: nombre correct d'octets		
Etiquette d'identificateur d'objet: 00000110		
Longueur d'identificateur d'objet: nombre correct d'octets		
Référence directe: toute syntaxe abstraite		
Etiquette de type ASN.1 individuel: 10100000		
Longueur de type ASN.1 individuel: nombre correct d'octets		
Certains octets de données utilisateur dans la syntaxe abstraite définie par l'utilisateur		

NUMÉRO DU TEST: 2.1.9.5.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Transport de l'information d'utilisateur; Accepter l'information d'utilisateur dans un message Continue		
OBJET: Vérifier qu'une IUT peut accepter un message Continue avec l'unité APDU 'AARE' comportant plusieurs éléments d'information d'utilisateurs		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos et les cas de test 2.1.9.5.2 doivent être réussis		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (CSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN (AARQ)</p> <p><i>ind. TR-CONTINUE</i></p> <p><=====</p>	<p>SP B (CSL)</p> <p>—————></p> <p>—————<</p> <p>CONTINUE (AARE)</p>	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue comportant des éléments d'information d'utilisateur dans la partie dialogue.	
3.	<p>VÉRIFICATION A: L'IUT ACCEPTE-T-ELLE L'INFORMATION D'UTILISATEUR DANS LA REPONSE DE DIALOGUE REÇUE?</p> <p>Faire également en sorte qu'un message END soit envoyé par l'IUT ou par le Testeur. Lorsque le dernier message a été envoyé par le Testeur, le message END envoyé par l'IUT peut être utilisé pour vérifier que le dernier message a bien été reçu.</p>	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011
 Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000
 Longueur externe: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: 00000111
 Référence directe: H'00118605010101
 Etiquette de type ASN.1 individuel: 10100000
 Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001
 Longueur de Réponse de Dialogue: nombre correct d'octets
 Etiquette de contexte d'application: 10100001
 Longueur de contexte d'application: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: nombre correct d'octets
 Référence directe: tout identificateur d'objet
 Etiquette Résultat: 10100010
 Longueur Résultat: 00000011
 Etiquette de type entier: 00000010
 Longueur d'entier: 00000001
 Valeur Résultat: 00000000 (Accepté)
 Etiquette de diagnostic d'origine du résultat: 10100011
 Longueur de diagnostic de l'origine du résultat: 00000101
 Etiquette de l'utilisateur du service de dialogue: 10100001
 Longueur de l'utilisateur du service de dialogue: 00000011
 Etiquette de type entier: 00000010
 Longueur d'entier: 00000001
 Valeur de l'utilisateur du service de dialogue: 00000000 (NUL)

Information d'utilisateur dans la dialogue PDU

Etiquette d'information d'utilisateur: 10111110
 Longueur d'information d'utilisateur: nombre correct d'octets
 Etiquette de type externe: 00101000 (nombre externe 1)
 Longueur externe: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: nombre correct d'octets
 Référence directe: toute syntaxe abstraite définie par l'utilisateur
 Etiquette de type ASN.1 individuel: 10100000
 Longueur de type ASN.1 individuel: nombre correct d'octets
 Certains octets représentant l'élément 1
 Etiquette de type externe: 00101000 (nombre externe 2)
 Longueur externe: nombre correct d'octets
 Etiquette d'identificateur d'objet: 00000110
 Longueur d'identificateur d'objet: nombre correct d'octets
 Référence directe: toute syntaxe abstraite définie par l'utilisateur
 Etiquette de type ASN.1 individuel: 10100000
 Longueur de type ASN.1 individuel: nombre correct d'octets
 Quelques octets de données utilisateur dans l'élément d'information d'utilisateur 2

NUMÉRO DU TEST: 2.1.9.6		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie dialogue		
SOUS-TITRE: Dialogue non structuré		
OBJET: Vérifier que l'IUT peut accepter l'unité APDU 'AUDT' de commande de dialogue dans un message Unidirectionnel		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p style="text-align: center;">←</p> <p><i>ind. TR-UNI</i> UNIDIRECTIONAL (AUDT)</p> <p>←=====</p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Unidirectionnel comportant une partie dialogue au point SP A.	
2.	VÉRIFICATION A: LE MESSAGE UNIDIRECTIONNEL AVEC L'UNITE APDU 'AUDT' A-T-IL ETE CORRECTEMENT REÇU AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
Partie dialogue dans le message Unidirectionnel Etiquette partie dialogue: 01101011 Longueur de partie dialogue: nombre correct d'octets Type de données externes dans la partie dialogue Etiquette de type externe: 00101000 Longueur externe: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: 00000111 Référence directe: H'00118605010201 Etiquette de type ASN.1 individuel: 10100000 Longueur de type ASN.1 individuel: nombre correct d'octets Dialogue PDU Etiquette de Demande de Dialogue: 01100000 Longueur de Demande de Dialogue: nombre correct d'octets Etiquette de contexte d'application: 10100001 Longueur de contexte d'application: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: nombre correct d'octets Référence directe: tout identificateur d'objet		

NUMÉRO DU TEST: 2.1.9.7.1		Feuillet 1 de 2
RÉFÉRENCE: 3.2.3/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Version unité APDU de commande de dialogue; Dialogue structuré; Version différente de 1		
OBJET: Vérifier qu'une IUT peut abandonner le dialogue si le premier bit du champ version du protocole dans la demande de dialogue, n'est pas réglé à 1. Le dialogue doit être abandonné avec l'unité APDU 'AARE' (rejet permanent, pas de dialogue commun)		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
	←	BEGIN (AARQ)
<i>dem. TR-UA-BORT</i>		
=====→		
ABORT (AARE)	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une demande de dialogue indiquant que la Version 1 n'est pas prise en charge	
2.	VÉRIFICATION A: LE POINT SP A A-T-IL TRANSMIS LE MESSAGE ABORT ATTENDU AVEC L'UNITE APDU 'AARE' INDIQUANT PAS DE PARTIE DIALOGUE COMMUNE?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Abort

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: mêmes octets que dans la demande de dialogue

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000001 (Rejet – Permanent)

Etiquette de diagnostic d'origine du résultat: 10100011

Longueur d'origine du résultat: 00000101

Etiquette du Fournisseur du service de dialogue: 10100010

Longueur du Fournisseur du service de dialogue: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur du Fournisseur de service du dialogue: 00000010 (Pas de partie dialogue commune)

NUMÉRO DU TEST 2.1.9.7.2		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.3/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Version unité APDU de commande de dialogue; Dialogue structuré; Version 1		
OBJET: Vérifier qu'une IUT peut accepter une demande de dialogue offrant plusieurs versions comportant la version 1. La réponse de l'IUT doit être en version 1		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p style="text-align: center;">←</p> <p><i>ind. TR-BEGIN</i> BEGIN (AARQ-Vx)</p> <p>←=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====></p> <p>CONTINUE (AARE-V1) →</p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une demande de dialogue offrant plusieurs versions comportant la Version 1.	
2.	VÉRIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE CONTINUE CONTIENT-ELLE L'UNITE APDU AARE ET EST-ELLE EN VERSION 1?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE		
Partie dialogue dans le message Begin Etiquette partie dialogue: 01101011 Longueur de partie dialogue: nombre correct d'octets Type de données externes dans la partie dialogue Etiquette de type externe: 00101000 Longueur externe: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: 00000111 Référence directe: H'00118605010101 Etiquette de type ASN.1 individuel: 10100000 Longueur de type ASN.1 individuel: nombre correct d'octets Dialogue PDU Etiquette de Demande de Dialogue: 01100000 Longueur de Demande de Dialogue: nombre correct d'octets Etiquette de Version de Protocole: 10000000 Longueur de Version du Protocole: 00000010 Version du Protocole: 00000110 11000000 (versions 1 et 2 prises en charge) Etiquette de contexte d'application: 10100001 Longueur de contexte d'application: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: nombre correct d'octets Référence directe: tout identificateur d'objet		

NUMÉRO DU TEST: 2.1.9.7.3		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Version unité APDU de commande de dialogue; Dialogue non structuré; Version différente de 1		
OBJET: Vérifier qu'une IUT peut ignorer un message UNIDIRECTIONAL si le premier bit du champ version du protocole dans la demande de dialogue AUDT, n'est pas réglé à 1		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos et le cas de test 2.1.9.6 doit être réussi		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL) <i>message ignoré</i>	←	SP B (CSL) UNIDIRECTIONAL (AUDT)
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Unidirectionnel comportant l'unité APDU AUDT indiquant que la Version 1 n'est pas prise en charge au point SP A.	
2.	VÉRIFICATION A: LE MESSAGE UNIDIRECTIONAL AVEC L'UNITE APDU 'AUDT' A-T-IL ETE IGNORE AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
Partie dialogue dans le message Unidirectionnel Etiquette partie dialogue: 01101011 Longueur de partie dialogue: nombre correct d'octets Type de données externes dans la partie dialogue Etiquette de type externe: 00101000 Longueur externe: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: 00000111 Référence directe: H'00118605010201 Etiquette de type ASN.1 individuel: 10100000 Longueur de type ASN.1 individuel: nombre correct d'octets Dialogue PDU Etiquette de Demande de Dialogue: 01100000 Longueur de Demande de Dialogue: nombre correct d'octets Etiquette de Version de Protocole: 10000000 Longueur de Version du Protocole: 00000010 Version du Protocole: 00000110 01000000 (seule la version 2 est prise en charge) Etiquette de contexte d'application: 10100001 Longueur de contexte d'application: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: nombre correct d'octets Référence directe: tout identificateur d'objet		

NUMÉRO DU TEST: 2.1.9.7.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Fonctions valides; Partie Dialogue		
SOUS-TITRE: Version unité APDU de commande de dialogue; Dialogue non structuré; Version 1		
OBJET: Vérifier qu'une IUT peut accepter un message UNIDIRECTIONAL si le premier bit de la version du protocole est réglé à 1 et que d'autres bits sont réglés à 1		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos et le cas de test 2.1.9.6 doit être réussi		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (CSL) SP B (CSL)</p> <p style="text-align: center;">← UNIDIRECTIONAL (AUDT)</p> <p><i>ind. TR-UNI</i></p> <p>←=====</p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Unidirectional comportant une unité APDU AUDT offrant plusieurs versions y compris la version 1, au point SP A.	
2.	VÉRIFICATION A: LE MESSAGE UNIDIRECTIONAL AVEC L'UNITE APDU AUDT' A-T-IL ETE ACCEPTE AU POINT SP A?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES

Partie dialogue dans le message Unidirectionnel

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010201

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Demande de Dialogue: 01100000

Longueur de Demande de Dialogue: nombre correct d'octets

Etiquette de Version du Protocole: 10000000

Longueur de Version du Protocole: 00000010

Version du Protocole: 00000110 11000000 (versions 1 et 2 prises en charge)

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: tout identificateur d'objet

NUMÉRO DU TEST: 2.2.1.1		Feuillet: 1 de 2															
RÉFÉRENCE: 6.2/Q.773																	
TITRE: Comportement syntaxiquement non valide; Valeurs non valides pour des éléments d'information.																	
SOUS-TITRE: Longueur d'identificateur d'invocation > 1 dans un composant d'invocation																	
OBJET: Vérifier qu'une opération demandée peut être rejetée du fait du codage incorrect du composant d'invocation (valeur hors tolérance)																	
CONDITIONS INITIALES: Faire en sorte que le stimulus soit tel qu'un message TSL approprié généré au point SP B comporte un composant d'invocation avec une erreur comme décrit ci-dessus																	
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP															
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%;">SP A (CSL)</td> <td style="width: 40%; text-align: center;">←</td> <td style="width: 30%;">SP B (CSL)</td> </tr> <tr> <td></td> <td></td> <td>INVOKE (i)</td> </tr> <tr> <td><i>ind. TC-L-REJECT</i></td> <td></td> <td></td> </tr> <tr> <td>←</td> <td></td> <td></td> </tr> <tr> <td>REJECT (NUL)</td> <td style="text-align: center;">→</td> <td></td> </tr> </table>			SP A (CSL)	←	SP B (CSL)			INVOKE (i)	<i>ind. TC-L-REJECT</i>			←			REJECT (NUL)	→	
SP A (CSL)	←	SP B (CSL)															
		INVOKE (i)															
<i>ind. TC-L-REJECT</i>																	
←																	
REJECT (NUL)	→																
DESCRIPTION DU TEST																	
1.	Initier une invocation d'opération du point SP B au point SP A avec un identificateur d'invocation de valeur égale à 2 octets (valeur illégale).																
2.	VÉRIFICATION A: LE COMPOSANT REJET AVEC UNE INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?																
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES																	
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie dialogue: nombre correct d'octets Composant d'INVOCATION dans un message TSL envoyé du point SP B au point SP A Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets Etiquette d'Identificateur d'invocation: 00000010 Longueur d'Identificateur d'invocation: 00000010 (deux octets) Identificateur d'invocation: 129																	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (local) ou 00000110 (global)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est d'une longueur d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur du TC)

Composant REJECT dans le message TSL envoyé du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets
Etiquette NULL: 00000101
Longueur NULL: 00000000
Etiquette de code de problème: 10000000 (type de problème général)
Longueur de code de problème: 00000001
Code de problème: 00000001 (composant incorrect)

NUMÉRO DU TEST: 2.2.1.2		Feuillet: 1 de 1
RÉFÉRENCE: 6.2/Q.773		
TITRE: Comportement syntaxiquement non valide; Valeurs non valides des éléments d'information		
SOUS-TITRE: Longueur de l'identificateur Invoke ID égale à 0 dans le composant Invoke		
OBJET: Vérifier qu'un rejet d'une opération demandée peut être effectué à cause d'un codage incorrect de l'ID de composant (longueur égale à 0)		
CONDITIONS INITIALES: Provoquer une stimulation telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke avec une erreur comme indiqué ci-dessous		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ←----- INVOKE <i>ind. TC-L-REJECT</i> ←===== REJECT (NULL) -----→ </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP B au point SP A avec un identificateur Invoke ID égal à 0 octet (valeur illégale).	
2.	VÉRIFICATION A: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets Composant INVOKE dans le message TSL du point SP B au point SP A Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000000 (zéro octet) Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) Code d'opération: x (x représente un code d'opération valide) paramètres (fournis par l'utilisateur-TC) Composant REJECT dans le message TSL du point SP A au point SP B Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets Etiquette NULL: 00000101 Longueur NULL: 00000000 Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)		

NUMÉRO DU TEST: 2.2.2.1.1		Feuillet: 1 de 1
RÉFÉRENCE: 6.2/Q.773		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Composant Invoke; identificateur Invoke ID manquant		
OBJET: Vérifier qu'un rejet d'une opération demandée peut être effectué à cause d'un identificateur Invoke ID manquant		
CONDITIONS INITIALES: Provoquer une stimulation telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke avec une erreur comme indiqué ci-dessous		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ← INVOKE ind. TC-L-REJECT <===== REJECT (NULL) → </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP B au point SP A avec un identificateur Invoke ID manquant.	
2.	VÉRIFICATION A: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets Composant INVOKE dans le message TSL du point SP B au point SP A Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets Etiquette de code d'opération: 00000110 (globale) Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) Code d'opération: x (x représente un code d'opération valide) paramètres (fournis par l'utilisateur-TC) Composant REJECT dans le message TSL du point SP A au point SP B Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets Etiquette NULL: 00000101 Longueur NULL: 00000000 Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)		

NUMÉRO DU TEST: 2.2.2.1.2		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Composant Invoke; Code d'opération manquant		
OBJET: Vérifier qu'un rejet d'une opération demandée peut être effectué à cause d'un code d'opération manquant		
CONDITIONS INITIALES: Provoquer une stimulation telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke avec une erreur de syntaxe comme indiqué ci-dessous		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ←----- INVOKE (i) ind. TC-L-REJECT <===== REJECT (i) -----> </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP B au point SP A avec code d'opération manquant.	
2.	VÉRIFICATION A: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT REJECT ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets Composant INVOKE dans le message TSL du point SP B au point SP A Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000001 (un octet) ID d'invocation: i (i représente un nombre entier) paramètres (fournis par l'utilisateur-TC) Composant REJECT dans le message TSL du point SP A au point SP B Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000001 ID d'invocation: i Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)		

NUMÉRO DU TEST: 2.2.2.2.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Composant Return Result; identificateur Invoke ID manquant		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause de l'absence de l'identificateur Invoke ID dans le composant Return Result-Last		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Last sans identificateur Invoke ID puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====> INVOKE (i) <i>ind. TC-L-REJECT</i> <=====	 —————> <————— —————>	SP B (CSL) RETURN RESULT-LAST
REJECT (NULL) expiration du temporisateur pour l'invocation (i)		
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 1 ou 3 du point SP A au point SP B. Créer une réponse du point SP B au point SP A sans identificateur Invoke ID.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: y (y est différent de x) (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette NULL: 00000101
 Longueur NULL: 00000000
 Etiquette de code de problème: 10000000 (type de problème général)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (composant type erroné)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.2.2.2.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Composant Return Result; Code d'opération manquant mais paramètres inclus		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause de l'absence du code d'opération dans le composant Return Result-Last		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke pour la Classe 1 ou 3</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Last sans code d'opération puisse être créé</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (i)</p>	<p>SP B (CSL)</p> <p>RETURN RESULT-LAST (i)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP A au point SP B. Créer une réponse du point SP B au point SP A avec un identificateur Invoke ID valide mais avec un code d'opération différent.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de séquence: 00110000
 Longueur de séquence: nombre correct d'octets
 paramètres (doivent être fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de code de problème: 10000000 (type de problème général)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (composant type erroné)

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (doivent être fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de code de problème: 10000000 (type de problème général)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (composant type erroné)

NUMÉRO DU TEST: 2.2.2.3.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Return Error; identificateur Invoke ID manquant		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause de l'absence de l'identificateur Invoke ID dans le composant Return Error		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke de Classe 1</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error sans un identificateur Invoke ID puisse être créé</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (NULL)</p> <p>expiration du temporisateur pour l'invocation (i)</p>	<p>SP B (CSL)</p> <p>RETURN ERROR</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 1 du point SP A au point SP B. Créer une réponse négative du point SP B au point SP A sans un identificateur Invoke ID.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN ERROR)
 Longueur de composant: nombre correct d'octets
 Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
 Code d'opération: y (y est un code d'erreur qui peut être signalé au titre de l'opération invoquée)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette NULL: 00000101
 Longueur NULL: 00000000
 Etiquette de code de problème: 10000000 (type de problème général)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (composant type erroné)

NUMÉRO DU TEST: 2.2.2.3.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Return Error; Code d'erreur manquant		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause de l'absence du code d'erreur dans le composant Return Error		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke de Classe 1 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error sans code d'erreur puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====> INVOKE (i)		SP B (CSL) RETURN ERROR (i)
<i>ind. TC-L-REJECT</i> <=====		
REJECT (i)		
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 1 du point SP A au point SP B. Créer une réponse négative du point SP B au point SP A avec un identificateur Invoke ID valide mais sans code d'erreur pour cette opération.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000101
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de code de problème: 10000000 (type de problème général)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (composant type erroné)

NUMÉRO DU TEST: 2.2.2.5.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Partie dialogue; Contexte d'application manquant dans l'APDU AARQ		
OBJET: Vérifier que l'IUT abandonne la transaction à la réception d'un message Begin comportant une unité APDU 'AARQ' sans paramètre de contexte d'application		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p><i>Détecter erreur de syntaxe</i> BEGIN (AARQ)</p> <p><i>dem. TR-U ABORT</i></p> <p>=====></p> <p>ABORT (ABRT) —————></p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une demande de dialogue sans contexte d'application.	
2.	VÉRIFICATION A: LE POINT SP A A-T-IL TRANSMIS LE MESSAGE ABORT ATTENDU AVEC L'UNITE APDU 'ABRT'?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE		
Partie dialogue dans le message Abort Etiquette partie dialogue: 01101011 Longueur de partie dialogue: 00010010 Type de données externes dans la partie dialogue Etiquette de type externe: 00101000 Longueur externe: 00010000 Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: 00000111 Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré) Etiquette de type ASN.1 individuel: 10100000 Longueur de type ASN.1 individuel: 00000101 Dialogue PDU Etiquette de dialogue abandonné: 01100100 Longueur de dialogue abandonné: 00000011 Etiquette d'origine de l'abandon: 10000000 Longueur de l'origine de l'abandon: 00000001 Origine Abandon: 00000001 (fournisseur de service du dialogue)		

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans message Begin

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Demande de Dialogue: 01100000

Longueur de Demande de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: manquante

NUMÉRO DU TEST: 2.2.2.5.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Partie Dialogue; longueur incorrecte		
OBJET: Vérifier que l'IUT abandonne la transaction à la réception d'un message Continue comportant une unité APDU 'AARE' avec un indicateur de longueur AARE incorrect		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN (AARQ) —————></p> <p><i>Détecter erreur de syntaxe</i> ←———— CONTINUE (AARE)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p> <p><i>dem. TR-U ABORT</i></p> <p>=====></p> <p>ABORT (ABRT) —————></p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue avec un indicateur de longueur AARE incorrect dans l'unité APDU AARE.	
3.	VERIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE ABORT CONTIENT-IL L'UNITE APDU 'ABRT'?	
4.	VERIFICATION B: VERIFIER QUE LE DIALOGUE AU POINT SP A A ETE TERMINE.	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: 01111111 (incorrect)

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

référence directe: mêmes octets que dans la demande de dialogue

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000000 (Accepté)

Etiquette de diagnostic d'origine du résultat: 10100011

Longueur d'origine du résultat: 00000101

Etiquette de l'utilisateur du service de dialogue: 10100001

Longueur de l'utilisateur du service de dialogue: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur de l'utilisateur du service de dialogue: 00000000 (Néant)

NUMÉRO DU TEST: 2.2.2.5.3		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Partie Dialogue; Diagnostic de l'origine du résultat manquant		
OBJET: Vérifier que l'IUT abandonne la transaction à la réception d'un message Continue comportant une unité APDU 'AARE' avec un paramètre de diagnostic d'origine du résultat manquant		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN (AARQ) —————></p> <p><i>Détecter erreur de syntaxe</i> ←———— CONTINUE (AARE)</p> <p><i>ind. TR-P-ABORT</i></p> <p><=====</p> <p><i>dem. TR-U-ABORT</i></p> <p>=====></p> <p>ABORT (ABRT) —————></p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue avec un AARE incorrect, paramètre manquant.	
3.	VERIFICATION A: VERIFIER QUE LE DIALOGUE AU POINT SP A A ETE TERMINE.	
4.	VERIFICATION B: LA PARTIE DIALOGUE DANS LE MESSAGE ABORT CONTIENT-IL L'UNITE APDU 'ABRT'?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: mêmes octets que dans la demande de dialogue

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000000 (Accepté)

Etiquette de diagnostic d'origine du résultat: manquante

NUMÉRO DU TEST: 2.2.2.5.4		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Partie Dialogue; Contexte d'application manquant dans l'APDU AUDT		
OBJET: Vérifier que l'IUT ignore un message Unidirectionnel comportant une demande de dialogue sans paramètre de contexte d'application		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL) <i>Détection d'une erreur de syntaxe</i>	←	SP B (CSL) UNIDIRECTIONAL (AUDT)
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Unidirectionnel avec la demande de dialogue endommagée au point SP A.	
2.	VÉRIFICATION A: VERIFIER QU'AUCUN MESSAGE N'EST GENERE EN REPONSE AU MESSAGE REÇU.	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE		
Partie dialogue dans le message Unidirectionnel Etiquette partie dialogue: 01101011 Longueur de partie dialogue: nombre correct d'octets Etiquette de type externe: 00101000 Longueur externe: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: 00000111 Référence directe: H'00118605010201 Etiquette de type ASN.1 individuel: 10100000 Longueur de type ASN.1 individuel: nombre correct d'octets Etiquette de Demande de Dialogue: 01100000 Longueur de Demande de Dialogue: nombre correct d'octets Etiquette de Version du Protocole: 10000000 Longueur de Version du Protocole: 00000010 Version du Protocole: 00000110 11000000 Etiquette de contexte d'application: manquante		

NUMÉRO DU TEST: 2.2.2.5.5		Feuillet: 1 de 1															
RÉFÉRENCE: 3.2.2.1/Q.774																	
TITRE: Comportement syntaxiquement non valide; Structure non valide																	
SOUS-TITRE: Partie Dialogue; Type externe sans référence directe																	
OBJET: Vérifier que l'IUT abandonne la transaction à la réception d'un message Begin comportant un type externe sans référence directe																	
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos																	
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP															
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%;">SP A (CSL)</td> <td style="width: 40%;"></td> <td style="width: 30%;">SP B (CSL)</td> </tr> <tr> <td><i>Détecter erreur sémantique</i></td> <td style="text-align: center;">←—————</td> <td>BEGIN (AARQ)</td> </tr> <tr> <td><i>dem. TR-U-ABORT</i></td> <td></td> <td></td> </tr> <tr> <td>=====></td> <td></td> <td></td> </tr> <tr> <td>ABORT (ABRT)</td> <td style="text-align: center;">—————→</td> <td></td> </tr> </table>			SP A (CSL)		SP B (CSL)	<i>Détecter erreur sémantique</i>	←—————	BEGIN (AARQ)	<i>dem. TR-U-ABORT</i>			=====>			ABORT (ABRT)	—————→	
SP A (CSL)		SP B (CSL)															
<i>Détecter erreur sémantique</i>	←—————	BEGIN (AARQ)															
<i>dem. TR-U-ABORT</i>																	
=====>																	
ABORT (ABRT)	—————→																
DESCRIPTION DU TEST																	
1.	Faire en sorte que le point SP B envoie un message Begin comportant une partie dialogue avec un type externe sans référence directe.																
2.	VÉRIFICATION A: LE POINT SP A A-T-IL TRANSMIS LE MESSAGE ABORT ATTENDU AVEC L'UNITE APDU 'ABRT'?																
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE																	
Partie dialogue dans le message Begin Etiquette partie dialogue: 01101011 Longueur de partie dialogue: nombre correct d'octets Type de données externes dans la partie dialogue Etiquette de type externe: 00101000 Longueur externe: nombre correct d'octets Etiquette d'identificateur d'objet: manquante Référence directe: manquante Etiquette de type ASN.1 individuel: 10100000 Longueur de type ASN.1 individuel: nombre correct d'octets TOUT nombre d'octets																	

NUMÉRO DU TEST: 2.2.2.5.6		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Partie Dialogue; Référence indirecte dans un type externe		
OBJET: Vérifier que l'IUT abandonne la transaction à la réception d'un message Begin comportant un type externe avec une référence directe et une référence indirecte		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>Détecter erreur sémantique</i>	←	BEGIN (AARQ)
<i>dem. TR-U-ABORT</i>		
=====→		
ABORT (ABRT)	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une partie dialogue avec un type externe comprenant également une référence indirecte.	
2.	VÉRIFICATION A: LE POINT SP A A-T-IL TRANSMIS LE MESSAGE ABORT ATTENDU AVEC L'UNITE APDU 'ABRT'?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE		
Partie dialogue dans le message Begin.		
Etiquette partie dialogue: 01101011		
Longueur de partie dialogue: nombre correct d'octets		
Type de données externes dans la partie dialogue		
Etiquette de type externe: 00101000		
Longueur externe: nombre correct d'octets		
Etiquette d'identificateur d'objet: 00000110		
Longueur d'identificateur d'objet: 00000111		
Référence directe: H'00118605010101		
Etiquette de type entier: 000000010		
Longueur d'entier: 00000001		
Référence directe: 00000001		
Etiquette de type ASN.1 individuel: 10100000		
Longueur de type ASN.1 individuel: nombre correct d'octets		
Dialogue PDU		
Etiquette de Demande de Dialogue: 01100000		
Longueur de Demande de Dialogue: nombre correct d'octets		
Etiquette de contexte d'application: 10100001		
Longueur de contexte d'application: nombre correct d'octets		
Etiquette d'identificateur d'objet: 00000110		
Longueur d'identificateur d'objet: nombre correct d'octets		
Référence directe: tout identificateur d'objet		

NUMÉRO DU TEST: 2.2.2.5.7		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Partie dialogue; Information d'utilisateur sans référence directe		
OBJET: Vérifier que l'IUT abandonne un dialogue à la réception d'un élément d'information d'utilisateur sans référence directe		
CONDITIONS INITIALES: Le point SP A (TSL) et et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>Détecter erreur sémantique</i>	←	BEGIN (AARQ)
<i>dem. TR-U-ABORT</i>		
=====→		
ABORT (ABRT)	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une partie dialogue avec un élément d'information d'utilisateur dont le type externe ne comporte pas de référence directe.	
2.	VÉRIFICATION A: LE POINT SP A A-T-IL TRANSMIS LE MESSAGE ABORT ATTENDU AVEC L'UNITE APDU 'ABRT'?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Begin

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Demande de Dialogue: 01100000

Longueur de Demande de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: tout identificateur d'objet

Information d'utilisateur dans le dialogue PDU

Etiquette d'information d'utilisateur: 10111110

Longueur d'information d'utilisateur: nombre correct d'octets

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: manquante

Longueur d'identificateur d'objet: manquante

Référence directe: manquante

Etiquette d'octet aligné: 10000001

Longueur d'octet aligné: nombre correct d'octets

Valeur de la chaîne d'octets: tout nombre d'octets

NUMÉRO DU TEST: 2.2.2.5.8		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Partie Dialogue; Référence indirecte dans une information d'utilisateur		
OBJET: Vérifier que l'IUT abandonne un dialogue à la réception d'un élément d'information d'utilisateur qui comporte une référence directe et une référence indirecte		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>Détecter erreur sémantique</i>	←	BEGIN (AARQ)
<i>dem. TR-U-ABORT</i>		
=====>		
ABORT (ABRT)	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une partie dialogue avec un type externe comprenant également une référence indirecte.	
2.	VÉRIFICATION A: LE POINT SP A A-T-IL TRANSMIS LE MESSAGE ABORT ATTENDU AVEC L'UNITE APDU 'ABRT'?	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Begin

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Demande de Dialogue: 01100000

Longueur de Demande de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: tout identificateur d'objet

Information d'utilisateur dans le dialogue PDU

Etiquette d'information d'utilisateur: 10111110

Longueur d'information d'utilisateur: nombre correct d'octets

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: tout identificateur d'objet

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Référence indirecte: 00000001

Etiquette d'octet aligné: 10000001

Longueur d'octet aligné: nombre correct d'octets

Valeur de la chaîne d'octets: tout nombre d'octets

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette non valide: 00100010

Longueur du code d'opération: nombre correct d'octets (par exemple, 00000001 si x est d'une longueur d'un octet)

Code d'opération: x (x représente un code d'opération valide)

Composant REJECT dans le message TSL envoyé du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'identificateur d'invocation: 00000010

Longueur d'Identificateur d'invocation: 00000001

Identificateur d'invocation: i

ou

Etiquette NULL: 00000101

Longueur NULL: 00000000

Etiquette de code de problème: 10000000 (type de problème général)

Longueur de code de problème: 00000001

Code de problème: 00000010 (composant mal structuré)

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
Code d'opération: x (x représente un code d'opération valide)
paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
Longueur d'ID d'invocation: 00000001
ID d'invocation: i

ou

Etiquette NULL: 00000101
Longueur NULL: 00000000

Etiquette de code de problème: 10000000 (type de problème général)
Longueur de code de problème: 00000001
Code de problème: 00000010 (composant mal structuré)

NUMÉRO DU TEST: 2.2.3.3		Feuille: 1 de 1
RÉFÉRENCE: 3.3/Q.773		
TITRE: Comportement syntaxiquement non valide; Codage non valide du composant Invoke		
SOUS-TITRE: EOC manquant dans la forme indéfinie		
OBJET: Vérifier qu'une partie composant avec une forme indéfinie mais avec un EOC manquant est rejetée		
CONDITIONS INITIALES: Provoquer une stimulation telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre> SP A (CSL) SP B (CSL) ←----- INVOKE (i) ind. TC-L-REJECT <===== REJECT (i ou NULL) -----> </pre>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération simple du point SP B au point SP A.	
2.	VÉRIFICATION A: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets Composant INVOKE dans le message TSL du point SP B au point SP A Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets (forme indéfinie) Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000001 (un octet) ID d'invocation: i (i représente un nombre entier) Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) Code d'opération: x (x représente un code d'opération valide) paramètres (fournis par l'utilisateur-TC) Composant REJECT dans le message TSL du point SP A au point SP B Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets Etiquette d'ID d'invocation: 00000010 Longueur d'ID d'invocation: 00000001 ID d'invocation: i ou Etiquette NULL: 00000101 Longueur NULL: 00000000 Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000010 (composant mal structuré)		

NUMÉRO DU TEST: 2.3.1.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement inopportun; Composant Invoke inopportun		
SOUS-TITRE: ID lié non valide		
OBJET: Vérifier qu'un rejet d'opération demandée peut être initialisé à cause d'un ID lié non valide		
CONDITIONS INITIALES:		
1) Provoquer une stimulation telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Invoke lié peut être créé comme indiqué ci-dessous		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) dem. TC-INVOKE</p> <p>=====></p> <p>INVOKE (i)</p> <p>ind. TC-L-REJECT</p> <p>=====</p> <p>REJECT (j) expiration du temporisateur pour l'invocation (i) ind. TC-L-CANCEL</p> <p>=====</p>	<p>SP B (CSL)</p> <p>INVOKE (j, k)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'IDENTIFICATEUR INVOKE ID DANS LE COMPOSANT REJECT ÉTAIT-IL IDENTIQUE À CELUI DU COMPOSANT INVOKE ENVOYÉ PAR LE POINT SP B?	
5.	VÉRIFICATION D: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant INVOKE dans le message TSL envoyé par le point SP B

Etiquette de type de composant: 10100001 (INVOKE)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001 (un octet)
 ID d'invocation: j (j représente un nombre entier)
 Etiquette d'ID lié: 10000000
 Longueur d'ID lié: 00000001 (un octet)
 ID lié: k (k est un nombre entier différent de i)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
 Code d'opération: y (y représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL envoyé par le point SP A

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: j
 Etiquette de code de problème: 10000001 (INVOKE)
 Longueur de code de problème: 00000001
 Code de problème: 00000101 (ID lié non reconnu)

NUMÉRO DU TEST: 2.3.2.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement inopportun; ID Invoke non reconnu		
SOUS-TITRE: Composant Return Result-Last inopportun		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause d'un identificateur Invoke ID non reconnu (qui n'a jamais été utilisé et vient tout juste d'être émis) dans le composant Return Result-Last reçu		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke pour une opération de Classe 1 ou 3</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Last avec un identificateur Invoke ID non valide puisse être créé</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (j)</p> <p>expiration du temporisateur pour l'invocation (i)</p> <p><i>ind. TC-L-CANCEL</i></p> <p><=====</p> <p><i>ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (i)</p>	<p>—————></p> <p>—————<</p> <p>—————></p> <p>—————<</p> <p>—————></p>	<p>SP B (CSL)</p> <p>RETURN RESULT-LAST (j)</p> <p>RETURN RESULT-LAST (i)</p>
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP A au point SP B. Créer une réponse du point SP B au point SP A avec un identificateur Invoke ID non reconnu.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	Créer un composant Return Result-Last du point SP B au point SP A.	
5.	VÉRIFICATION C: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
6.	VÉRIFICATION D: LE FLUX DE COMPOSANTS ÉTAIT-IL COMME INDIQUÉ CI-DESSUS?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j (j est différent de i)

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j

Etiquette de code de problème: 10000010 (RETURN RESULT)

Longueur de code de problème: 00000001

Code de problème: 00000000 (identificateur Invoke ID non reconnu)

Le contenu des deux derniers composants RETURN RESULT-LAST (i) et REJECT (i) est le même que celui indiqué ci-dessus pour ces composants, sauf que l'identificateur Invoke ID est égal à (i).

NOTE – Omis lorsque aucun paramètre n'est présent.

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT NOT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j (j est différent de i)

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000000 (identificateur Invoke ID non reconnu)

Composant RETURN RESULT NOT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000000 (identificateur Invoke ID non reconnu)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.3.2.3		Feuillet: 1 de 3
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement inopportun; ID Invoke non reconnu		
SOUS-TITRE: Composant Return Error inopportun		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause d'un identificateur Invoke ID non reconnu (qui n'a jamais été utilisé et vient tout juste d'être émis) dans le composant Return Error reçu		
CONDITIONS INITIALES:		
<p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke pour une opération de Classe 1 ou 2 non reconnue</p> <p>2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error avec identificateur Invoke ID non valide puisse être créé</p>		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (j)</p> <p>expiration du temporisateur pour l'invocation (i)</p> <p><i>ind. TC-L-CANCEL</i></p> <p><=====</p> <p><i>ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (i)</p>	<p>SP B (CSL)</p> <p>RETURN ERROR (j)</p> <p>RETURN ERROR (i)</p>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP A au point SP B. Créer une réponse négative du point SP B au point SP A avec un identificateur Invoke ID non valide.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	Créer un composant Return Error du point SP B au point SP A.	
5.	VÉRIFICATION C: LE COMPOSANT REJECT A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
6.	VÉRIFICATION D: LE FLUX DE COMPOSANTS ÉTAIT-IL COMME INDIQUÉ CI-DESSUS?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j (j est différent de i)

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'erreur: y

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j

Etiquette de code de problème: 10000011 (RETURN ERROR)

Longueur de code de problème: 00000001

Code de problème: 00000000 (identificateur Invoke ID non reconnu)

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Composant RETURN ERROR dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)

Code d'erreur: y

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de code de problème: 10000011 (RETURN ERROR)

Longueur de code de problème: 00000001

Code de problème: 00000000 (identificateur Invoke ID non reconnu)

NUMÉRO DU TEST: 2.3.2.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement inopportun; ID Invoke non reconnu		
SOUS-TITRE: Composant Reject inopportun		
OBJET: Vérifier que la réception d'un composant Reject avec un identificateur Invoke ID ne correspondant à aucune invocation active n'a pas d'effet sur une invocation active		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke pour une opération de Classe 1 ou 2 2) Faire en sorte que les données au point SP B soient telles qu'un composant Reject avec un identificateur Invoke ID non reconnu puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i) <i>ind. TC-R-REJECT a)</i> <=====	—————→ ←—————	REJECT (j)
 <i>ind. TC-L-RESULT</i> <=====	←—————	RETURN RESULT-LAST (i)
a) L'émission de l'indication TC-R-REJECT dépend du mode d'implémentation utilisé.		
DESCRIPTION DU TEST		
1.	Invocation d'une opération du point SP A au point SP B. Créer un composant Reject du point SP B au point SP A avec un identificateur Invoke ID non valide.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	Créer un composant Reject du point SP B au point SP A.	
4.	VÉRIFICATION B: LE FLUX DE COMPOSANTS ÉTAIT-IL COMME INDIQUÉ CI-DESSUS?	
5.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Partie composant dans les messages TSL

Etiquette de partie composant: 01101100

Longueur de partie composant: nombre correct d'octets

Composant INVOKE dans le message TSL du point SP A au point SP B

Etiquette de type composant: 10100001 (INVOKE)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001 (un octet)

ID d'invocation: i (i représente un nombre entier)

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)

Code d'opération: x (x représente un code d'opération valide)

paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: j (j est différent de i)

Etiquette de code de problème: 10000001 (INVOKE)

Longueur de code de problème: 00000001

Code de problème: valeur quelconque

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)

Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010

Longueur d'ID d'invocation: 00000001

ID d'invocation: i

Etiquette de séquence: 00110000 (voir la Note)

Longueur de séquence: nombre correct d'octets (voir la Note)

Etiquette de code d'opération: 00000010 (locale) ou 00000101 (globale) (voir la Note)

Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)

Code d'opération: x (voir la Note)

paramètres (fournis par l'utilisateur-TC)

NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.3.3.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Comportement inopportun; Composants imprévus		
SOUS-TITRE: Return Result-Last pour la Classe 2		
OBJET: Vérifier qu'un rejet peut être initialisé si un composant Return Result-Last est reçu pour une opération de Classe 2		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————>	
	<—————	RETURN RESULT-LAST (i)
<i>ind. TC-L-REJECT</i> <=====		
REJECT (i)	—————>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 2 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (return result imprévu)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.3.3.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Comportement inopportun; Composants imprévus		
SOUS-TITRE: Return Result-Last pour la Classe 4		
OBJET: Vérifier qu'un rejet peut être envoyé si un composant Return Result-Last est reçu pour une opération de Classe 4		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————>	
	<—————	RETURN RESULT-LAST (i)
<i>ind. TC-L-REJECT</i> <=====		
REJECT (i)	—————>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 4 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100010 (RETURN RESULT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (return result imprévu)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.3.3.3		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Comportement inopportun; Composants imprévus		
SOUS-TITRE: Return Result Not-Last pour la Classe 2		
OBJET: Vérifier qu'un rejet peut être envoyé si un composant Return Result Not-Last est reçu pour une opération de Classe 2		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Not-Last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————>	
	<—————	RETURN RESULT NOT-LAST (i)
<i>ind. TC-L-REJECT</i> <=====		
REJECT (i)	—————>	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 2 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT NOT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (return result imprévu)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.3.3.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Comportement inopportun; Composants imprévus		
SOUS-TITRE: Return Result Not-Last pour la Classe 4		
OBJET: Vérifier qu'un rejet peut être envoyé si un composant Return Result Not-Last est reçu pour une opération de Classe 4		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke		
2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result-Not-Last puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
<p>SP A (CSL) SP B (CSL)</p> <p><i>dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) RETURN RESULT NOT-LAST (i)</p> <p style="text-align: center;">—————→</p> <p style="text-align: center;">←————</p> <p><i>ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (i) —————→</p>		
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 4 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN RESULT NOT-LAST dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de séquence: 00110000 (voir la Note)
 Longueur de séquence: nombre correct d'octets (voir la Note)
 Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) (voir la Note)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet) (voir la Note)
 Code d'opération: x (voir la Note)
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i
 Etiquette de code de problème: 10000010 (RETURN RESULT)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (return result imprévu)
 NOTE – Omis lorsque aucun paramètre n'est présent.

NUMÉRO DU TEST: 2.3.3.5		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Comportement inopportun; Composants imprévus		
SOUS-TITRE: Return Error pour la Classe 3		
OBJET: Vérifier qu'un rejet peut être envoyé si un composant Return Error est reçu pour une opération de Classe 3		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————→	
	←—————	RETURN ERROR (i)
<i>ind. TC-L-REJECT</i> <=====		
REJECT (i)	—————→	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 3 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur-TC)

Composant RETURN ERROR dans le message TSL du point SP B au point SP A

Etiquette de type de composant: 10100011 (RETURN ERROR)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale)
 Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si y est long d'un octet)
 Code d'erreur: y
 paramètres (fournis par l'utilisateur-TC)

Composant REJECT dans le message TSL du point SP A au point SP B

Etiquette de type de composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets

Etiquette d'ID d'invocation: 00000010
 Longueur d'ID d'invocation: 00000001
 ID d'invocation: i

Etiquette de code de problème: 10000011 (RETURN ERROR)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (return error imprévu)

NUMÉRO DU TEST: 2.3.3.6		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Comportement inopportun; Composants imprévus		
SOUS-TITRE: Return Error pour la Classe 4		
OBJET: Vérifier qu'un rejet peut être envoyé si un composant Return Error est reçu pour une opération de Classe 4		
CONDITIONS INITIALES:		
1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne un composant Invoke 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Error puisse être créé		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>dem. TC-INVOKE</i> =====>		SP B (CSL)
INVOKE (i)	—————→	
	←—————	RETURN ERROR (i)
<i>ind. TC-L-REJECT</i> <=====		
REJECT (i)	—————→	
DESCRIPTION DU TEST		
1.	Invocation d'une opération de Classe 4 du point SP A au point SP B.	
2.	VÉRIFICATION A: LE COMPOSANT INVOKE AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
3.	VÉRIFICATION B: LE COMPOSANT REJECT AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?	
4.	VÉRIFICATION C: L'AUTOMATE D'INVOCATION ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
Partie composant dans les messages TSL		
Etiquette de partie composant: 01101100		
Longueur de partie composant: nombre correct d'octets		
Composant INVOKE dans le message TSL du point SP A au point SP B		
Etiquette de type de composant: 10100001 (INVOKE)		
Longueur de composant: nombre correct d'octets		
Etiquette d'ID d'invocation: 00000010		
Longueur d'ID d'invocation: 00000001 (un octet)		
ID d'invocation: i (i représente un nombre entier)		

TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES

Etiquette de code d'opération: 00000010 (local) ou 00000110 (global)
 Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est d'une longueur d'un octet)
 Code d'opération: x (x représente un code d'opération valide)
 paramètres (fournis par l'utilisateur TC)

COMPOSANT RETURN ERROR dans un message TSL envoyé du point SP B au point SP A

Etiquette de type composant: 10100011 (RETURN ERROR)
 Longueur de composant: nombre correct d'octets
 Etiquette d'Identificateur d'invocation: 00000010
 Longueur d'Identificateur d'invocation: 00000001
 Identificateur d'invocation: i
 Etiquette de code d'erreur: 00000010 (local) ou 00000110 (global)
 Longueur de code erreur: nombre correct d'octets (par exemple, 00000001 si y est une longueur d'un octet)
 Code erreur: y
 paramètres (fournis par l'utilisateur TC)

Composant REJECT dans le message TSL envoyé du point SP A au point SP B

Etiquette de type composant: 10100100 (REJECT)
 Longueur de composant: nombre correct d'octets
 Etiquette d'Identificateur d'invocation: 00000010
 Longueur d'Identificateur d'invocation: 00000001
 Identificateur d'invocation: i
 Etiquette de code de problème: 10000011 (RETURN ERROR)
 Longueur de code de problème: 00000001
 Code de problème: 00000001 (return error non attendu)

NUMÉRO DU TEST: 2.3.4.1		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1.2/Q.774		
TITRE: Comportement inopportun; Partie Dialogue, unités APDU imprévues		
SOUS-TITRE: Message Begin avec une APDU AARE		
OBJET: Vérifier qu'une IUT abandonne la transaction à la réception d'un message Begin comportant une partie dialogue qui transporte une unité APDU 'AARE'		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>Détecter erreur sémantique</i>	←	BEGIN (AARE)
<i>dem. TR-U-ABORT</i>		
=====>		
ABORT (U) (ABRT)	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une réponse de dialogue.	
2.	VÉRIFICATION A: LE POINT SP A A-T-IL TRANSMIS LE MESSAGE ABORT ATTENDU AVEC L'UNITE APDU 'ABRT'?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE		
Partie dialogue dans le message Abort		
Etiquette partie dialogue: 01101011		
Longueur de partie dialogue: nombre correct d'octets		
Type de données externes dans la partie dialogue		
Etiquette de type externe: 00101000		
Longueur externe: nombre correct d'octets		
Etiquette d'identificateur d'objet: 00000110		
Longueur d'identificateur d'objet: 00000111		
Référence directe: H'00118605010101		
Etiquette de type ASN.1 individuel: 10100000		
Longueur de type ASN.1 individuel: nombre correct d'octets		
Dialogue PDU		
Etiquette de dialogue abandonné: 01100100		
Longueur de dialogue abandonné: 00000011		
Etiquette d'origine de l'abandon: 10000000		
Longueur de l'origine de l'abandon: 00000001		
Origine Abandon: 00000001 (fournisseur de service du dialogue)		

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans Begin

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: 00011001

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: tout identificateur d'objet

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000000 (Accepté)

Etiquette de diagnostic d'origine du résultat: 10100011

Longueur de diagnostic de l'origine du résultat: 00000101

Etiquette de l'utilisateur du service de dialogue: 10100001

Longueur de l'utilisateur du service de dialogue: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur de l'utilisateur du service de dialogue: 00000000 (Néant)

NUMÉRO DU TEST: 2.3.4.2		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement inopportun; Partie Dialogue, unités APDU imprévues		
SOUS-TITRE: Confirmation de Dialogue avec une APDU autre que AARE		
OBJET: Vérifier qu'une IUT abandonne la transaction à la réception d'un message Continue comportant une partie dialogue qui transporte une unité APDU 'AARQ'		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>dem. TR-BEGIN</i>		
=====>		
BEGIN (AARQ)	—————>	
<i>Détecter erreur sémantique</i>	<—————	CONTINUE (AARQ)
<i>ind. TR-P-ABORT</i>		
<=====		
<i>dem. TR-U-ABORT</i>		
=====>		
ABORT (U) (ABRT)	—————>	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue avec une partie dialogue comportant une unité AARQ APDU.	
3.	VERIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE ABORT CONTIENT-ELLE UNE UNITE APDU 'ABRT'?	
4.	VERIFICATION B: VERIFIER QUE LE DIALOGUE AU POINT SP A A ETE TERMINE.	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Demande de Dialogue: 01100000

Longueur de Demande de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: tout identificateur d'objet

Partie dialogue dans le message Abort

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: 00010010

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: 00010000

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: 00000101

Dialogue PDU

Etiquette de dialogue abandonné: 01100100

Longueur de dialogue abandonné: 00000011

Etiquette d'origine de l'abandon: 10000000

Longueur de l'origine de l'abandon: 00000001

Origine Abandon: 00000001 (fournisseur de service du dialogue)

NUMÉRO DU TEST: 2.3.4.3		Feuille: 1 de 2
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement inopportun; Partie Dialogue, unités APDU imprévues		
SOUS-TITRE: Confirmation de Dialogue avec une APDU ABRT		
OBJET: Vérifier qu'une IUT abandonne la transaction à la réception d'un message Continue comportant une partie dialogue qui transporte une unité 'ABRT' APDU.		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>dem. TR-BEGIN</i>		
=====>		
BEGIN (AARQ)	—————>	
<i>Détecter erreur sémantique</i>	←—————	CONTINUE (ABRT)
<i>ind. TR-P-ABORT</i>		
<=====		
<i>dem. TR-U-ABORT</i>		
=====>		
ABORT (U) (ABRT)	—————>	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue avec une partie dialogue comportant une unité ABRT APDU.	
3.	VERIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE ABORT CONTIENT-ELLE UNE UNITE APDU 'ABRT'? (se reporter au test numéro 2.3.4.2 Table de vérification pour ABRT).	
4.	VERIFICATION B: VERIFIER QUE LE DIALOGUE AU POINT SP A A ETE TERMINE.	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: 00010010

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: 00010000

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: 00000101

Dialogue PDU

Etiquette de dialogue abandonné: 01100100

Longueur de dialogue abandonné: 00000011

Etiquette d'origine de l'abandon: 10000000

Longueur de l'origine de l'abandon: 00000001

Origine Abandon: 00000001 (fournisseur de service du dialogue)

NUMÉRO DU TEST: 2.3.4.4		Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement inopportun, Partie Dialogue, unités APDU imprévues		
SOUS-TITRE: Présence d'une partie dialogue APDU à l'état actif		
OBJET: Vérifier qu'une IUT abandonne la transaction à la réception d'un message Continue comportant une partie dialogue qui transporte une unité 'AARE' APDU		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos et le cas de test 2.1.9.1.2 doit être réussi		
CONFIGURATION: 1	TYPE DE TEST: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (CSL)</p> <p><i>ind. TR-BEGIN</i></p> <p>=====</p> <p><i>dem. TR-CONTINUE</i></p> <p>=====></p> <p>CONTINUE (AARE)</p> <p><i>ind. TR-P-ABORT</i></p> <p>=====</p> <p><i>dem. TR-U-ABORT</i></p> <p>=====></p> <p>ABORT (U) (ABRT)</p>	<p>←</p> <p>←</p> <p>→</p> <p>←</p> <p>→</p>	<p>SP B (CSL)</p> <p>BEGIN (AARQ)</p> <p>CONTINUE (AARE)</p>
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin avec la demande de dialogue au point SP A.	
2.	Faire en sorte que le point SP A confirme le dialogue.	
3.	Faire en sorte que le point SP B envoie un message Continue comportant une partie dialogue transportant une unité AARE APDU.	
4.	VÉRIFICATION A: VERIFIER QUE L'IUT AU POINT SP A TERMINE LA TRANSACTION.	

TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE

Partie dialogue dans le message Continue

Etiquette partie dialogue: 01101011

Longueur de partie dialogue: nombre correct d'octets

Type de données externes dans la partie dialogue

Etiquette de type externe: 00101000

Longueur externe: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: 00000111

Référence directe: H'00118605010101

Etiquette de type ASN.1 individuel: 10100000

Longueur de type ASN.1 individuel: nombre correct d'octets

Dialogue PDU

Etiquette de Réponse de Dialogue: 01100001

Longueur de Réponse de Dialogue: nombre correct d'octets

Etiquette de contexte d'application: 10100001

Longueur de contexte d'application: nombre correct d'octets

Etiquette d'identificateur d'objet: 00000110

Longueur d'identificateur d'objet: nombre correct d'octets

Référence directe: mêmes octets que dans AARQ

Etiquette Résultat: 10100010

Longueur Résultat: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur Résultat: 00000000 (Accepté)

Etiquette de diagnostic d'origine du résultat: 10100011

Longueur de diagnostic de l'origine du résultat: 00000101

Etiquette de l'utilisateur du service de dialogue: 10100001

Longueur de l'utilisateur du service de dialogue: 00000011

Etiquette de type entier: 00000010

Longueur d'entier: 00000001

Valeur de l'utilisateur du service de dialogue: 00000000 (Néant)

NUMÉRO DU TEST: 2.3.4.5		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement inopportun; Partie Dialogue, unités APDU imprévues		
SOUS-TITRE: Message Unidirectionnel avec une syntaxe abstraite non attendue		
OBJET: Vérifier qu'une IUT ignore un message UNIDIRECTIONAL comportant une partie dialogue faisant référence à une syntaxe abstraite non attendue		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p>SP A (CSL) SP B (CSL)</p> <p><i>Détection d'une erreur</i> ← UNIDIRECTIONAL (AARQ)</p>		
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Unidirectionnel comportant une partie dialogue mais faisant référence à la syntaxe abstraite de dialogue structuré.	
2.	VÉRIFICATION A: LE POINT SP A IGNORE T-IL LE MESSAGE?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES		
Partie dialogue dans le message Unidirectionnel Etiquette partie dialogue: 01101011 Longueur de partie dialogue: nombre correct d'octets Type de données externes dans la partie dialogue Etiquette de type externe: 00101000 Longueur externe: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: 00000111 Référence directe: H'00118605010101 (syntaxe abstraite de dialogue structuré) Etiquette de type ASN.1 individuel: 10100000 Longueur de type ASN.1 individuel: nombre correct d'octets Dialogue PDU Etiquette de Demande de Dialogue: 01100000 Longueur de Demande de Dialogue: nombre correct d'octets Etiquette de contexte d'application: 10100001 Longueur de contexte d'application: nombre correct d'octets Etiquette d'identificateur d'objet: 00000110 Longueur d'identificateur d'objet: nombre correct d'octets Référence directe: tout identificateur d'objet		

NUMÉRO DU TEST: 2.3.4.6		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement inopportun; Partie Dialogue, unités APDU imprévues		
SOUS-TITRE:., Partie dialogue non attendue dans le message Continue		
OBJET: Vérifier qu'une IUT abandonne la transaction à la réception d'un message Continue comportant une partie dialogue alors qu'aucune partie dialogue n'a été envoyée dans le message Begin		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>dem. TR-BEGIN</i>		
=====>		
BEGIN	—————>	
<i>Détecter erreur</i>	←—————	CONTINUE (AARE)
<i>ind. TR-P-ABORT</i>		
<=====		
<i>dem. TR-U-ABORT</i>		
=====>		
ABORT (U) (ABRT)	—————>	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin (sans partie dialogue)	
2.	Faire en sorte que le point SP B confirme le dialogue avec une partie dialogue comportant une unité AARE APDU. (Se reporter à la table de vérification du test numéro 2.3.4.4.)	
3.	VERIFICATION A: LA PARTIE DIALOGUE DANS LE MESSAGE ABORT CONTIENT-ELLE UNE UNITE APDU 'ABRT'? (Se reporter à la table de vérification du test numéro 2.3.4.2.)	
4.	VERIFICATION B: VERIFIER QUE LE DIALOGUE AU POINT SP A A ETE TERMINE.	

NUMÉRO DU TEST: 2.3.4.7		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement inopportun; Partie Dialogue, unités APDU imprévues		
SOUS-TITRE: Partie dialogue manquante dans le message Continue		
OBJET: Vérifier qu'une IUT abandonne la transaction à la réception d'un message Continue sans partie dialogue alors qu'une partie dialogue a été envoyée dans le message Begin		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>dem. TR-BEGIN</i>		
=====>		
BEGIN (AARQ)	—————>	
<i>Détecter erreur</i>	←—————	CONTINUE
<i>ind. TR-P-ABORT</i>		
<=====		
<i>dem. TR-U-ABORT</i>		
=====>		
ABORT (U) (ABRT)	—————>	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP A envoie un message Begin comportant une partie dialogue.	
2.	Faire en sorte que le point SP B confirme le dialogue avec un message Continue sans partie dialogue.	
3.	VERIFICATION A: VERIFIER QUE LE DIALOGUE AU POINT SP A A ETE TERMINE.	

NUMÉRO DU TEST: 2.3.4.8		Feuillet: 1 de 1
RÉFÉRENCE: 3.2.2.1/Q.774		
TITRE: Comportement inopportun; Partie Dialogue, unités APDU imprévues		
SOUS-TITRE: Message Begin avec syntaxe abstraite non attendue		
OBJET: Vérifier qu'une IUT abandonne la transaction à la réception d'un message Begin comportant une partie dialogue faisant référence à une syntaxe abstraite non attendue		
CONDITIONS INITIALES: Le point SP A (TSL) et le point SP B (TSL) doivent être à l'état de repos		
CONFIGURATION: 1	TYPE DE TEST: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
SP A (CSL)		SP B (CSL)
<i>Détecter erreur</i>	←	BEGIN (AUDT)
<i>dem. TR-U-ABORT</i>		
=====→		
ABORT (U) (ABRT)	→	
DESCRIPTION DU TEST		
1.	Faire en sorte que le point SP B envoie un message Begin comportant une partie dialogue faisant référence à une syntaxe abstraite non reconnue.	
2.	VÉRIFICATION A: LE POINT SP A A-T-IL TERMINE LA TRANSACTION?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LA PARTIE DIALOGUE		
Partie dialogue dans Begin		
Etiquette partie dialogue: 01101011		
Longueur de partie dialogue: nombre correct d'octets		
Type de données externes dans la partie dialogue		
Etiquette de type externe: 00101000		
Longueur externe: nombre correct d'octets		
Etiquette d'identificateur d'objet: 00000110		
Longueur d'identificateur d'objet: 00000111		
Référence directe: H'00118605010201 (syntaxe abstraite non structurée)		
Etiquette de type ASN.1 individuel: 10100000		
Longueur de type ASN.1 individuel: nombre correct d'octets		
Dialogue PDU		
Etiquette de Demande de Dialogue: 01100000		
Longueur de Demande de Dialogue: nombre correct d'octets		
Etiquette de contexte d'application: 10100001		
Longueur de contexte d'application: nombre correct d'octets		
Etiquette d'identificateur d'objet: 00000110		
Longueur d'identificateur d'objet: nombre correct d'octets		
Référence directe: tout identificateur d'objet		

SERIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série B	Moyens d'expression: définitions, symboles, classification
Série C	Statistiques générales des télécommunications
Série D	Principes généraux de tarification
Série E	Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains
Série F	Services de télécommunication non téléphoniques
Série G	Systèmes et supports de transmission, systèmes et réseaux numériques
Série H	Systèmes audiovisuels et multimédias
Série I	Réseau numérique à intégration de services
Série J	Transmission des signaux radiophoniques, télévisuels et autres signaux multimédias
Série K	Protection contre les perturbations
Série L	Construction, installation et protection des câbles et autres éléments des installations extérieures
Série M	RGT et maintenance des réseaux: systèmes de transmission, de télégraphie, de télécopie, circuits téléphoniques et circuits loués internationaux
Série N	Maintenance: circuits internationaux de transmission radiophonique et télévisuelle
Série O	Spécifications des appareils de mesure
Série P	Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux
Série Q	Commutation et signalisation
Série R	Transmission télégraphique
Série S	Equipements terminaux de télégraphie
Série T	Terminaux des services télématiques
Série U	Commutation télégraphique
Série V	Communications de données sur le réseau téléphonique
Série X	Réseaux pour données et communication entre systèmes ouverts
Série Y	Infrastructure mondiale de l'information
Série Z	Langages de programmation