

Remplacée par une version plus récente



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

Q.787

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

(03/93)

**SPÉCIFICATIONS DU SYSTÈME
DE SIGNALISATION N° 7**

**SPÉCIFICATION D'ESSAIS POUR
LES CAPACITÉS DE TRANSACTION**

Recommandation UIT-T Q.787

Remplacée par une version plus récente

(Antérieurement «Recommandation du CCITT»)

Remplacée par une version plus récente

AVANT-PROPOS

L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'Union internationale des télécommunications (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 que les Commissions d'études de l'UIT-T doivent examiner et à propos desquels elles doivent émettre des Recommandations.

La Recommandation révisée UIT-T Q.787, élaborée par la Commission d'études XI (1988-1993) de l'UIT-T, a été approuvée par la CMNT (Helsinki, 1-12 mars 1993).

NOTES

1 Suite au processus de réforme entrepris au sein de l'Union internationale des télécommunications (UIT), le CCITT n'existe plus depuis le 28 février 1993. Il est remplacé par le Secteur de la normalisation des télécommunications de l'UIT (UIT-T) créé le 1^{er} mars 1993. De même, le CCIR et l'IFRB ont été remplacés par le Secteur des radiocommunications.

Afin de ne pas retarder la publication de la présente Recommandation, aucun changement n'a été apporté aux mentions contenant les sigles CCITT, CCIR et IFRB ou aux entités qui leur sont associées, comme «Assemblée plénière», «Secrétariat», etc. Les futures éditions de la présente Recommandation adopteront la terminologie appropriée reflétant la nouvelle structure de l'UIT.

2 Dans la présente Recommandation, le terme «Administration» désigne indifféremment une administration de télécommunication ou une exploitation reconnue.

© UIT 1994

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.

Remplacée par une version plus récente

TABLE DES MATIÈRES

	<i>Page</i>
1 Introduction	1
2 Objectifs de la spécification d'essais	1
3 Champ d'application	1
4 Principes généraux des essais	2
5 Contexte d'exécution des essais	2
5.1 Relation sémaphore	2
5.2 Configuration	2
6 Trafic de base	2
7 Liste d'essais	3
7.1 Spécification d'essais de la sous-couche transaction TC	3
7.2 Spécification d'essais de la sous-couche composant TC	115

Remplacée par une version plus récente

Recommandation Q.787

SPÉCIFICATION D'ESSAIS POUR LES CAPACITÉS DE TRANSACTION

(Helsinki, 1993)

1 Introduction

La présente Recommandation contient un ensemble détaillé d'essais des capacités de transaction (TC) (*transaction capabilities*) du système de signalisation n° 7. Ces essais ont pour but de valider le protocole spécifié dans les Recommandations Q.771 à Q.774. La présente Recommandation est conforme au *Livre bleu* (1988) qui décrit les règles de base d'une spécification d'essais, telles qu'elles sont énoncées dans la Recommandation Q.780.

2 Objectifs de la spécification d'essais

La spécification d'essais 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 bleu* (1988) pour les capacités TC du SS n° 7.

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

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

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

3 Champ d'application

Les descriptions d'essai se divisent en deux parties: 7.1 essais de la sous-couche transaction (TSL) (*transaction sublayer*) et 7.2 essais de la sous-couche composant (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 essais et d'en faciliter la compréhension; elle n'implique aucune mise en œuvre particulière.

La présente spécification d'essais 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 essais décrits dans la présente Recommandation nécessitent la création de primitives; lors de l'exécution de ces essais, 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.

L'essai 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 essais.

=====> = PRIMITIVE

-----> = MESSAGE

Remplacée par une version plus récente

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

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

Les «Directives» 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 d'essais est indépendante de toute application ou mise en œuvre particulière.

4 Principes généraux des essais

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

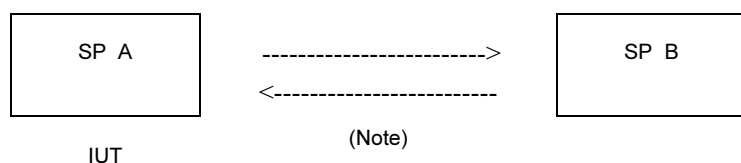
5 Contexte d'exécution des essais

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 essais de compatibilité.

5.2 Configuration

Une seule configuration, représentée dans la Figure 1, est nécessaire pour effectuer les essais indiqués dans la liste d'essais proposée.



NOTE – Les flèches indiquent une relation sémaphore.

FIGURE 1/Q.787

Configuration: 1

6 Trafic de base

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

Remplacée par une version plus récente

7 Liste d'essais

Les catégories de la liste d'essais sont indiquées dans les paragraphes ci-dessous.

7.1 Spécification d'essais de la sous-couche transaction TC

7.1.1 Directives relatives à l'exécution des essais de la sous-couche transaction

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

Dans la séquence de messages attendue, les primitives sont indiquées au point SP A [côté instance 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 d'essai. 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 essais des variations de longueur. Cependant, différentes formes répondant aux conditions du 3.3/Q.773, peuvent être utilisées, quel que soit l'essai.

Il est suggéré d'utiliser la procédure suivante pour vérifier les résultats antérieurs et postérieurs aux essais (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, ce qui indique que l'automate est à l'état de repos. Si le message Continue est accepté par l'IUT et suivi d'une réponse correcte, cela signifie que l'automate était à l'état initialisation envoyée.

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

7.1.2 Essais de la sous-couche transaction

NDA Aucun détail disponible (*no details available*)

FFS Pour étude ultérieure (*for further study*)

* = Validation et Compatibilité

Tous les autres essais sont des essais 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 de la transaction (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

Remplacée par une version plus récente

- 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
(Pour étude ultérieure)
- 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
 - 1.1.3.1.2 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 (Etablissement)
 - 1) Longueur d'OTID (Identificateur de transaction d'origine = 0
 - 2) Longueur d'OTID > quatre octets

Remplacée par une version plus récente

- 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 Unidirectional
 - 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
 - 1) OTID dupliqué
 - 2) 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 incorporés
 - 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

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.1.1	Feuillet: 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 D'ESSAI: VAT y CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; margin-top: 20px;"> <div style="text-align: center;"> <p>SP A (TSL)</p> <p><i>Dem. TR-UNI</i></p> <p>=====→</p> <p>UNIDIRECTIONAL -----→</p> </div> <div style="text-align: center;"> <p>SP B (TSL)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
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 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <p>SP A (TSL)</p> <p style="margin-left: 40px;"><i>Ind. TR-BEGIN</i> ←=====</p> <p style="margin-left: 40px;"><i>Dem. TR-END</i> =====→ (Terminaison de base)</p> <p>END</p> </div> <div style="width: 10%; text-align: center;"> <p>-----</p> <p>-----</p> <p>-----</p> </div> <div style="width: 45%;"> <p>SP B (TSL)</p> <p>BEGIN</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.1.2.1 1)	Feuillet: 2 de 2
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 destination: 01001000 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	
END	
Etiquette de type de message: 01100100 Longueur de type de message: nombre correct d'octets	
Etiquette d'ID de transaction de départ: 01001001 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 de l'OTID reçue dans le message BEGIN)	
Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="text-align: center;"> <p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i></p> <p><=====</p> <p><i>Dem. TR-END</i></p> <p>=====></p> <p><i>(Terminaison prédéterminée)</i></p> </div> <div style="text-align: center; flex-grow: 1;"> <p><-----></p> <p>BEGIN</p> </div> <div style="text-align: center;"> <p>SP B (TSL)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
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	
<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>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.1.2.1 3)	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; Emission IUT; Abandon par l'utilisateur-TR	
OBJET: Vérifier que le point sémaphore A peut recevoir un message Begin 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
SÉQUENCE DE MESSAGES ATTENDUE:	
SP A (TSL)	SP B (TSL)
	<----- BEGIN
<i>Ind. TR-BEGIN</i> <=====	
<i>Dem. TR-U-ABORT</i> =====>	
ABORT (U)	----->
DESCRIPTION DE L'ESSAI	
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-U-ABORT 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: UN MESSAGE ABORT A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?
5.	VÉRIFICATION C: LE DTID DANS LE MESSAGE ABORT 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 destination: 01001000	
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 d'information abandon par l'utilisateur: 01101100	
Longueur d'information abandon par l'utilisateur: 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 départ: 01001001	
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 de l'OTID reçue dans le message BEGIN)	
Etiquette de partie composant: 01101011	
Longueur de partie composant: nombre correct d'octets	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <pre style="font-family: monospace;"> SP A (TSL) SP B (TSL) Dem. TR-BEGIN =====> BEGIN -----> <----- <----- ABORT (U) Ind. TR-U-ABORT <===== </pre>	
DESCRIPTION DE L'ESSAI	
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	
<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 (U)</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 d'information abandon par l'utilisateur: 01101011 Longueur d'information abandon par l'utilisateur: nombre correct d'octets</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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 (TSL)</p> <p><i>Dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> </td> <td style="width: 40%; vertical-align: top; text-align: center;"> <p>-----></p> <p><-----</p> </td> <td style="width: 30%; 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><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><i>Ind. TR-P-ABORT</i></p> <p><=====</p>	<p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>ABORT (P)</p>		
DESCRIPTION DE L'ESSAI				
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: NOMBRE ENTIER (entre 0 et 4)</p>				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p><i>Ind. TR-END</i></p> <p><=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (TSL)</p> <p>-----></p> <p><-----</p> <p>END</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
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		
<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: 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 partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1.1 1)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>-----></p> <p><i>Ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>Dem. TR-END</i></p> <p>=====></p> <p><i>(Terminaison prédéterminée)</i></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (TSL)</p> <p>CONTINUE</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1.1 2)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
SÉQUENCE DE MESSAGES ATTENDUE:	
<p style="text-align: center;">SP A (TSL)</p> <p style="text-align: center;"><i>Dem. TR-BEGIN</i> =====></p> <p style="text-align: center;">BEGIN</p> <p style="text-align: center;"><i>Ind. TR-CONTINUE</i> <=====</p> <p style="text-align: center;"><i>Dem. TR-U-ABORT</i> =====></p> <p style="text-align: center;">ABORT (U)</p>	<p style="text-align: center;">SP B (TSL)</p> <p style="text-align: center;">-----></p> <p style="text-align: center;"><-----</p> <p style="text-align: center;">CONTINUE</p> <p style="text-align: center;">-----></p>
DESCRIPTION DE L'ESSAI	
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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1.1 3)	Feuille: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 30%;"> <p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE</p> <p><i>Ind. TR-END</i> <=====</p> </div> <div style="width: 30%; text-align: center;"> <p><-----</p> <p>-----></p> <p><-----</p> </div> <div style="width: 30%; text-align: right;"> <p>SP B (TSL)</p> <p>BEGIN</p> <p>END</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1.2 1)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 Libération 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 D'ESSAI: 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)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE</p> <p><i>Ind. TR-P-ABORT</i> <=====</p> </td> <td style="width: 40%; border: none; text-align: center; vertical-align: middle;"> <p><-----</p> <p>-----></p> <p><-----</p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (TSL)</p> <p>BEGIN</p> <p>ABORT (P)</p> </td> </tr> </table>		<p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE</p> <p><i>Ind. TR-P-ABORT</i> <=====</p>	<p><-----</p> <p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>ABORT (P)</p>
<p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE</p> <p><i>Ind. TR-P-ABORT</i> <=====</p>	<p><-----</p> <p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>ABORT (P)</p>		
DESCRIPTION DE L'ESSAI				
<ol style="list-style-type: none"> 1. 2. 3. 4. 5. 6. 7. 	<p>Faire en sorte que le point SP B envoie un message Begin au point SP A.</p> <p>Faire en sorte que le point SP A réponde par un message Continue.</p> <p>Faire en sorte que le point SP B réponde par un message Abort (P).</p> <p>VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?</p> <p>VÉRIFICATION B: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?</p> <p>VÉRIFICATION C: LE MESSAGE ABORT A-T-IL ÉTÉ REÇU CORRECTEMENT AU POINT SP A?</p> <p>VÉRIFICATION D: LES AUTOMATES TSL ASSOCIÉS À CETTE TRANSACTION ONT-ILS ÉTÉ LAISSÉS À L'ÉTAT DE REPOS AU POINT SP A?</p>			

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1.2 2)	Feuillet: 2 de 2
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: NOMBRE ENTIER (0 à 4)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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 (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE</p> <p><i>Ind. TR-U-ABORT</i> <=====</p> </td> <td style="width: 40%; border: none; text-align: center; vertical-align: middle;"> <p><-----</p> <p>-----></p> <p><-----</p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (TSL)</p> <p>BEGIN</p> <p>ABORT (U)</p> </td> </tr> </table>		<p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE</p> <p><i>Ind. TR-U-ABORT</i> <=====</p>	<p><-----</p> <p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>ABORT (U)</p>
<p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE</p> <p><i>Ind. TR-U-ABORT</i> <=====</p>	<p><-----</p> <p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>ABORT (U)</p>		
DESCRIPTION DE L'ESSAI				
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?			

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1.2.3)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1 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 initialisation; 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE -----></p> <p><i>Dem. TR-END</i> =====> (Terminaison de base)</p> <p>END -----></p>	<p>SP B (TSL)</p> <p>BEGIN</p>	
DESCRIPTION DE L'ESSAI		
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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.2.1 1)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1 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 initialisation; 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (TSL)</p> <p style="margin-left: 40px;"><i>Ind. TR-BEGIN</i> ←=====</p> <p style="margin-left: 40px;"><i>Dem. TR-CONTINUE</i> =====→</p> <p style="margin-left: 40px;">CONTINUE</p> <p style="margin-left: 40px;"><i>Dem. TR-END</i> =====→</p> <p style="margin-left: 40px;"><i>(Terminaison prédéterminée)</i></p> </div> <div style="width: 10%; text-align: center;"> <p>-----></p> <p>-----></p> </div> <div style="width: 45%;"> <p>SP B (TSL)</p> <p style="margin-left: 40px;">BEGIN</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.1 2)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.2.1 3)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.2.2.1)	Feillet: 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 D'ESSAI: 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 (TSL)</p> <p><i>Dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>-----></p> <p><i>Ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p> </td> <td style="width: 40%; border: none; text-align: center;"> <p>-----></p> <p><-----</p> <p>-----></p> </td> <td style="width: 30%; vertical-align: top; border: none;"> <p>SP B (TSL)</p> <p>CONTINUE</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>-----></p> <p><i>Ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p>	<p>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>END</p>
<p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>-----></p> <p><i>Ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p>	<p>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>END</p>			
DESCRIPTION DE L'ESSAI					
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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.2.2.1)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 TSL					
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 D'ESSAI: 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)</p> <p><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Ind. TR-P-ABORT</i> <=====</p> </td> <td style="width: 40%; border: none; text-align: center; vertical-align: top;"> <p>-----></p> <p><-----</p> <p>-----></p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</p> <p>ABORT (P)</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Ind. TR-P-ABORT</i> <=====</p>	<p>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>ABORT (P)</p>
<p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Ind. TR-P-ABORT</i> <=====</p>	<p>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>ABORT (P)</p>			
DESCRIPTION DE L'ESSAI					
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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.2.2.2)	Feuillet: 2 de 2
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: NOMBRE ENTIER (0 à 4)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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 (TSL)</p> <p><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Ind. TR-U-ABORT</i> <=====</p> </td> <td style="width: 40%; border: none; vertical-align: top;"> <p style="text-align: center;">-----></p> <p style="text-align: center;"><-----</p> <p style="text-align: center;"><-----</p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</p> <p>ABORT (U)</p> </td> </tr> </table>		<p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Ind. TR-U-ABORT</i> <=====</p>	<p style="text-align: center;">-----></p> <p style="text-align: center;"><-----</p> <p style="text-align: center;"><-----</p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>ABORT (U)</p>
<p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Ind. TR-U-ABORT</i> <=====</p>	<p style="text-align: center;">-----></p> <p style="text-align: center;"><-----</p> <p style="text-align: center;"><-----</p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>ABORT (U)</p>		
DESCRIPTION DE L'ESSAI				
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?			

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.2.2.2 3)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.3.1	Feuillet: 2 de 2
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 (1 à 4) 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>CONTINUE</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 (1 à 4) Valeur d'ID de transaction de départ: CHAÎNE D'OCTETS (longueur: 1 à 4 octets)</p> <p>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)</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: 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)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p><i>Ind. TR-END</i> <=====</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p><-----</p> <p>-----></p> <p><-----</p> <p>-</p> </td> <td style="width: 40%; vertical-align: top;"> <p>SP B (TSL)</p> <p>BEGIN</p> <p>CONTINUE</p> <p>END</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></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>CONTINUE</p> <p>END</p>
<p>SP A (TSL)</p> <p><i>Ind. TR-BEGIN</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></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>CONTINUE</p> <p>END</p>			
DESCRIPTION DE L'ESSAI					
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 E: L'AUTOMATE TSL A-T-IL ÉTÉ LAISSÉ À L'ÉTAT DE REPOS AU POINT SP A?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.3.2	Feuillet: 2 de 2
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 (1 à 4) Valeur d'ID de transaction de départ: CHAÎNE D'OCTETS (longueur: 1 à 4 octets)</p> <p>CONTINUE</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 (1 à 4) Valeur d'ID de transaction de départ: CHAÎNE D'OCTETS (longueur: 1 à 4 octets)</p> <p>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)</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: 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)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT	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><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></p> <p>CONTINUE</p> <p><i>Ind. TR-END</i> <=====</p> </td> <td style="width: 40%; vertical-align: top; text-align: center;"> <p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</p> <p>END</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></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>CONTINUE</p> <p>END</p>
<p>SP A (TSL)</p> <p><i>Dem. TR-BEGIN</i> =====></p> <p>BEGIN</p> <p><i>Ind. TR-CONTINUE</i> <=====</p> <p><i>Dem. TR-CONTINUE</i> =====></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>CONTINUE</p> <p>END</p>			
DESCRIPTION DE L'ESSAI					
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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.4.1	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP																																							
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; text-align: center;">SP A (TSL)</td> <td style="width: 40%;"></td> <td style="width: 30%; text-align: center;">SP B (TSL)</td> </tr> <tr> <td></td> <td style="text-align: center;"><-----</td> <td style="text-align: center;">BEGIN</td> </tr> <tr> <td style="text-align: center;"><i>Ind. TR-BEGIN</i></td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;"><=====</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;"><i>Dem. TR-CONTINUE</i></td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">=====></td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">CONTINUE</td> <td style="text-align: center;">-----></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;"><-----</td> <td style="text-align: center;">CONTINUE</td> </tr> <tr> <td style="text-align: center;"><i>Ind. TR-CONTINUE</i></td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;"><=====</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;"><i>Ind. TR-END</i></td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">=====></td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">END</td> <td style="text-align: center;">-----></td> <td></td> </tr> </table>			SP A (TSL)		SP B (TSL)		<-----	BEGIN	<i>Ind. TR-BEGIN</i>			<=====			<i>Dem. TR-CONTINUE</i>			=====>			CONTINUE	----->			<-----	CONTINUE	<i>Ind. TR-CONTINUE</i>			<=====			<i>Ind. TR-END</i>			=====>			END	----->	
SP A (TSL)		SP B (TSL)																																							
	<-----	BEGIN																																							
<i>Ind. TR-BEGIN</i>																																									
<=====																																									
<i>Dem. TR-CONTINUE</i>																																									
=====>																																									
CONTINUE	----->																																								
	<-----	CONTINUE																																							
<i>Ind. TR-CONTINUE</i>																																									
<=====																																									
<i>Ind. TR-END</i>																																									
=====>																																									
END	----->																																								
DESCRIPTION DE L'ESSAI																																									
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?																																								

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.4.2	Feuillet: 2 de 2
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: 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	
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: 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)	
Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.2.5	Feuillet: 1 de 1
RÉFÉRENCE:	
TITRE: Adressage TC (Pour étude ultérieure)	

NUMÉRO DE L'ESSAI: 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 dans la forme courte	
OBJET: Vérifier que le point sémaphore A peut accepter un message Begin dont la longueur est codée à l'aide de la forme courte 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 D'ESSAI: VAT
TYPE DE SP: SP	
SÉQUENCE DE MESSAGES ATTENDUE:	
SP A (TSL)	SP B (TSL)
	<----- BEGIN
<i>Ind. TR-BEGIN</i> <=====	
<i>Dem. TR-END</i> =====> (Terminaison de base)	
END	----->
DESCRIPTION DE L'ESSAI	
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» de l'essai
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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.3.1.1.1 1)	Feuillet: 2 de 2
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 courte 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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT
TYPE DE SP: SP	
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <p style="text-align: center;">SP A (TSL)</p> <p style="text-align: center;"><i>Ind. TR-BEGIN</i> <=====</p> <p style="text-align: center;"><i>Dem. TR-END</i> =====> (Terminaison de base)</p> <p style="text-align: center;">END</p> </div> <div style="width: 10%; text-align: center;"> <p>-----></p> <p>-----></p> <p>-----></p> </div> <div style="width: 45%;"> <p style="text-align: center;">SP B (TSL)</p> <p style="text-align: center;">BEGIN</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
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» de l'essai.
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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.3.1.1.1 2)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 courte 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 D'ESSAI: VAT
TYPE DE SP: SP	
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="text-align: center;"> <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> </div> <div style="text-align: center; flex-grow: 1;"> <p><-----</p> <p>BEGIN</p> <p>-----></p> </div> <div style="text-align: center;"> <p>SP B (TSL)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
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» de l'essai.
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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.3.1.1.2 1)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT
TYPE DE SP: SP	
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <p style="text-align: center;">SP A (TSL)</p> <p style="text-align: center;"><i>Ind. TR-BEGIN</i> <=====</p> <p style="text-align: center;"><i>Dem. TR-END</i> =====></p> <p style="text-align: center;">(Terminaison de base)</p> <p style="text-align: center;">END</p> </div> <div style="width: 10%; text-align: center;"> <p><-----</p> <p>-----></p> </div> <div style="width: 40%; text-align: right;"> <p>SP B (TSL)</p> <p>BEGIN</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
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» de l'essai.
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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.3.1.1.3 1)	Feuillet: 2 de 2
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 (FIN DE DONNÉES): 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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <p>SP A (TSL)</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></p> <p>END</p> </div> <div style="width: 10%; text-align: center;"> <p><-----</p> <p>-----></p> </div> <div style="width: 40%; align-self: center;"> <p>SP B (TSL)</p> <p>BEGIN</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.3.2.1 1)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (TSL)</p> <p style="margin-left: 100px;"><i>Ind. TR-BEGIN</i></p> <p style="margin-left: 100px;"><=====</p> <p style="margin-left: 100px;"><i>Dem. TR-END</i></p> <p style="margin-left: 100px;">=====></p> <p style="margin-left: 100px;">(Terminaison de base)</p> <p>END</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (TSL)</p> <p>BEGIN</p> <p>-----></p> </div> </div>		
DESCRIPTION DE L'ESSAI		
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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.1.3.2.1 2)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer 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 un OTID d'une longueur de 0 octet								
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP						
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; text-align: center;">SP A (TSL)</td> <td style="width: 40%;"></td> <td style="width: 30%; text-align: center;">SP B (TSL)</td> </tr> <tr> <td style="text-align: center;"><i>:Détection de l'erreur de syntaxe</i></td> <td style="text-align: center;"><-----</td> <td style="text-align: center;">BEGIN</td> </tr> </table>			SP A (TSL)		SP B (TSL)	<i>:Détection de l'erreur de syntaxe</i>	<-----	BEGIN
SP A (TSL)		SP B (TSL)						
<i>:Détection de l'erreur de syntaxe</i>	<-----	BEGIN						
DESCRIPTION DE L'ESSAI								
1.	Faire en sorte que le point SP B envoie le message Begin erroné, avec un OTID long de 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								
<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: 0 octet Valeur d'ID de transaction de départ: non présente</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>								

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.1.1 2)	Feillet: 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 D'ESSAI: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p style="text-align: center;"> SP A (TSL) SP B (TSL) </p> <p style="text-align: center;"> <i>:Détection de l'erreur de syntaxe</i> <----- BEGIN </p>		
DESCRIPTION DE L'ESSAI		
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		
<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: cinq octets Valeur d'ID de transaction de départ: CHAÎNE D'OCTETS (longueur: 5 octets)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer 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 octet			
CONFIGURATION: 1	TYPE D'ESSAI: VAT		
TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; 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) (voir Note) -----></p> </td> <td style="width: 50%; vertical-align: top; text-align: right;"> <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) (voir Note) -----></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) (voir Note) -----></p>	<p>SP B (TSL)</p> <p>CONTINUE</p>		
<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.</p>			
DESCRIPTION DE L'ESSAI			
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?		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.1.2 1)	Feuillet: 2 de 2
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>CONTINUE</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 (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 (1 à 4 octets) (valeur de l'OTID reçue dans le message CONTINUE)</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</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.1.3 1)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer 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 A soient telles que la longueur du DTID du message END soit supérieure à quatre octets	
CONFIGURATION: 1	TYPE D'ESSAI: VAT
TYPE DE SP: SP	
SÉQUENCE DE MESSAGES ATTENDUE:	
SP A (TSL)	SP B (TSL)
<i>Dem. TR-BEGIN</i> =====>	
BEGIN	----->
<i>:Détection de l'erreur</i>	<----- END
DESCRIPTION DE L'ESSAI	
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 invalide) 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>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 interdite)		
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <div style="width: 45%;"> <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><i>Ind. TR-P-ABORT</i></p> <p><=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (TSL)</p> <p>-----></p> <p><-----</p> <p>ABORT (P)</p> </div> </div>		
NOTE – L'envoi de l'indication TR-Abort dépend du mode de mise en œuvre utilisé.		
DESCRIPTION DE L'ESSAI		
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 interdite).	
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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.1.5 1)	Feuillet: 2 de 2
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: nombre correct d'octets Valeur de cause P-Abort: NOMBRE ENTIER (5 – valeur interdite pour ce champ)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.1.5 2)	Feuille: 2 de 2
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: NOMBRE ENTIER (0 à 4)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer 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 D'ESSAI: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p style="text-align: center;"> SP A (TSL) SP B (TSL) </p> <p style="text-align: center;"> <i>:Détection de l'erreur de syntaxe</i> <----- UNIDIRECTIONAL </p>		
DESCRIPTION DE L'ESSAI		
1.	Faire en sorte que le point SP B envoie un message unidirectionnel 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		
<p>UNIDIRECTIONAL</p> <p>Etiquette de type de message: 01100001</p> <p>Longueur de type de message: nombre correct d'octets</p> <p>Partie composant manquante</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer 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 D'ESSAI: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p style="text-align: center;"> SP A (TSL) SP B (TSL) </p> <p style="text-align: center;"> <i>:Détection de l'erreur de syntaxe</i> <----- BEGIN </p>		
DESCRIPTION DE L'ESSAI		
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		
<p>BEGIN</p> <p>Etiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets</p> <p>OTID absent</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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; Élément 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 supprimer 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 D'ESSAI: 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>:Détection de l'erreur de syntaxe</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)	:Détection de l'erreur de syntaxe	<-----	BEGIN	ABORT (P)	----->	
SP A (TSL)		SP B (TSL)									
:Détection de l'erreur de syntaxe	<-----	BEGIN									
ABORT (P)	----->										
<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.</p>											
DESCRIPTION DE L'ESSAI											
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?										

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.2 2)	Feuillet: 2 de 2
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 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</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>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer 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 D'ESSAI: 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>Dem. TR-BEGIN</i></td> <td></td> <td></td> </tr> <tr> <td><u>=====</u>></td> <td></td> <td></td> </tr> <tr> <td>BEGIN</td> <td style="text-align: center;">-----></td> <td></td> </tr> <tr> <td><i>:Détection de l'erreur de syntaxe</i></td> <td style="text-align: center;"><-----</td> <td>CONTINUE</td> </tr> </table>			SP A (TSL)		SP B (TSL)	<i>Dem. TR-BEGIN</i>			<u>=====</u> >			BEGIN	----->		<i>:Détection de l'erreur de syntaxe</i>	<-----	CONTINUE
SP A (TSL)		SP B (TSL)															
<i>Dem. TR-BEGIN</i>																	
<u>=====</u> >																	
BEGIN	----->																
<i>:Détection de l'erreur de syntaxe</i>	<-----	CONTINUE															
DESCRIPTION DE L'ESSAI																	
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 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: 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?																

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.3 1)	Feuillet: 2 de 2
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>CONTINUE</p> <p>Etiquette de type de message: 01100101 Longueur de type de message: nombre correct d'octets</p> <p>OTID absent</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 partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.3 2)	Feillet: 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 supprimer 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 D'ESSAI: 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><i>:Détection de l'erreur de syntaxe</i></p> <p>ABORT (P)</p> </td> <td style="width: 30%; vertical-align: top; border: none;"> <p>-----></p> <p><-----</p> <p>-----></p> </td> <td style="width: 35%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</p> <p>(DTID absent)</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>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>(DTID absent)</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>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>(DTID absent)</p>			
<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.</p>					
DESCRIPTION DE L'ESSAI					
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 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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.3 2)	Feuillet: 2 de 2
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>CONTINUE</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)</p> <p>DTID absent</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 d'OTID reçue dans le message CONTINUE)</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>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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'instance 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 D'ESSAI: 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 style="text-align: center;">(avec OTID dupliqué)</p> <p><-----</p> <p>ABORT (P)</p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> </td> <td style="width: 30%; vertical-align: top; border: none;"> <p>-----></p> <p style="text-align: center;">(avec OTID dupliqué)</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>-----></p> <p style="text-align: center;">(avec OTID dupliqué)</p> <p><-----</p> <p>ABORT (P)</p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p>	<p>-----></p> <p style="text-align: center;">(avec OTID dupliqué)</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>-----></p> <p style="text-align: center;">(avec OTID dupliqué)</p> <p><-----</p> <p>ABORT (P)</p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p>	<p>-----></p> <p style="text-align: center;">(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 DE L'ESSAI					
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 PAR LE POINT SP A?				
4.	VÉRIFICATION B: SI UN MESSAGE ABORT A ÉTÉ ENVOYÉ, A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE 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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.3 3)	Feuille: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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'instance 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 D'ESSAI: 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 style="text-align: center;">(avec DTID dupliqué)</p> <p><-----</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>-----></p> <p style="text-align: center;">(avec DTID dupliqué)</p> <p><-----</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>-----></p> <p style="text-align: center;">(avec DTID dupliqué)</p> <p><-----</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>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.</p>					
DESCRIPTION DE L'ESSAI					
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 PAR LE 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 PAR LE 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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.3 4)	Feuille: 2 de 2
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'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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.3 5)	Feuillet: 1 de 2				
RÉFÉRENCE: 3.3.4/Q.774					
TITRE: Comportement syntaxiquement non valide; Structure non valide					
SOUS-TITRE: Premier message Continue; Élément d'information inconnu présent					
OBJET: Vérifier que, à la réception d'un message Continue erroné, 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 Continue avec un OTID déductible et un DTID déductible et affecté contienne une erreur de syntaxe et soit envoyé au SP A en réponse au message Begin					
CONFIGURATION: 1	TYPE D'ESSAI: 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><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; border: none;"> <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>			
<p>NOTE – Si le message et la primitive ABORT ne sont pas pas envoyés, il peut s'agir d'un comportement valide, ce qui dépend du mode de mise en œuvre utilisé.</p>					
DESCRIPTION DE L'ESSAI					
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 avec un élément d'information supplémentaire après l'élément d'information DTID (par exemple, cause P-Abort).				
3.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE CONTINUE AU POINT SP A.				
4.	VÉRIFICATION B: 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? (PARTIE TRANSACTION INCORRECTE)				
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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.3 5)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer le message					
CONDITIONS INITIALES: Les points SP A (TSL) et SP B (TSL) doivent être à l'état de repos					
CONFIGURATION: 1	TYPE D'ESSAI: 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><i>Ind. TR-CONTINUE</i></p> <p><=====</p> <p><i>:Détection de l'erreur</i></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> </table>			<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>
<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 DE L'ESSAI					
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 PAR LE 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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.4 1)	Feuille: 2 de 2
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 ^{ème} 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)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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)</p> <p style="text-align: center;"><i>Ind. TR-BEGIN</i> <=====</p> <p style="text-align: center;"><i>Dem. TR-CONTINUE</i> =====></p> <p style="text-align: center;">:Détection de l'erreur</p> <p>ABORT (P)</p> <p style="text-align: center;"><i>Ind. TR-P-ABORT</i> <=====</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p><-----</p> <p>-----></p> <p><-----</p> <p>-----></p> </td> <td style="width: 40%; vertical-align: top;"> <p>SP B (TSL)</p> <p>BEGIN</p> <p>CONTINUE</p> <p>CONTINUE</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p style="text-align: center;"><i>Ind. TR-BEGIN</i> <=====</p> <p style="text-align: center;"><i>Dem. TR-CONTINUE</i> =====></p> <p style="text-align: center;">:Détection de l'erreur</p> <p>ABORT (P)</p> <p style="text-align: center;"><i>Ind. TR-P-ABORT</i> <=====</p>	<p><-----</p> <p>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>CONTINUE</p> <p>CONTINUE</p>
<p>SP A (TSL)</p> <p style="text-align: center;"><i>Ind. TR-BEGIN</i> <=====</p> <p style="text-align: center;"><i>Dem. TR-CONTINUE</i> =====></p> <p style="text-align: center;">:Détection de l'erreur</p> <p>ABORT (P)</p> <p style="text-align: center;"><i>Ind. TR-P-ABORT</i> <=====</p>	<p><-----</p> <p>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>BEGIN</p> <p>CONTINUE</p> <p>CONTINUE</p>			
<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.</p>					
DESCRIPTION DE L'ESSAI					
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 (Elément 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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.4 2)	Feuillet: 2 de 2
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 ^{ème} 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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer 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 D'ESSAI: VAT
TYPE DE SP: SP	
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p style="margin-left: 40px;">SP A (TSL) SP B (TSL)</p> <p style="margin-left: 40px;"><i>Dem. TR-BEGIN</i></p> <p style="margin-left: 40px;">=====></p> <p style="margin-left: 40px;">BEGIN -----></p> <p style="margin-left: 40px;">:Détection de l'erreur de syntaxe <----- END</p>	
DESCRIPTION DE L'ESSAI	
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	
<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>DTID absent</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimer 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 D'ESSAI: VAT	TYPE DE SP: SP
SÉQUENCE DE MESSAGES ATTENDUE:		
<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>		
DESCRIPTION DE L'ESSAI		
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		
<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>DTID absent</p> <p>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</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.7 1)	Feillet: 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 supprimer 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 D'ESSAI: VAT	TYPE DE SP: SP						
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 35%; text-align: center;">SP A (TSL)</td> <td style="width: 30%;"></td> <td style="width: 35%; text-align: center;">SP B (TSL)</td> </tr> <tr> <td style="text-align: center;"><i>:Détection du type de message Unknown</i></td> <td style="text-align: center;">←-----</td> <td style="text-align: center;">MESSAGE UNKNOWN</td> </tr> </table>			SP A (TSL)		SP B (TSL)	<i>:Détection du type de message Unknown</i>	←-----	MESSAGE UNKNOWN
SP A (TSL)		SP B (TSL)						
<i>:Détection du type de message Unknown</i>	←-----	MESSAGE UNKNOWN						
DESCRIPTION DE L'ESSAI								
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								

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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)</p> <p><i>:Détection du type de message Unknown</i></p> <p>ABORT (P)</p> </td> <td style="width: 30%; vertical-align: middle; text-align: center;"> <p><-----</p> <p>-----></p> </td> <td style="width: 40%; vertical-align: top;"> <p>SP B (TSL)</p> <p>MESSAGE UNKNOWN</p> </td> </tr> </table>			<p>SP A (TSL)</p> <p><i>:Détection du type de message Unknown</i></p> <p>ABORT (P)</p>	<p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>MESSAGE UNKNOWN</p>
<p>SP A (TSL)</p> <p><i>:Détection du type de message Unknown</i></p> <p>ABORT (P)</p>	<p><-----</p> <p>-----></p>	<p>SP B (TSL)</p> <p>MESSAGE UNKNOWN</p>			
<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.</p>					
DESCRIPTION DE L'ESSAI					
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					
<p>MESSAGE UNKNOWN</p> <p>Étiquette de type de message: unknown (par exemple, 01100110)</p> <p>Longueur de type de message: nombre correct d'octets</p> <p>Étiquette d'ID de transaction de départ: 01001000</p> <p>Longueur d'ID de transaction de départ: nombre correct d'octets</p> <p>Valeur d'ID de transaction de départ: CHAÎNE D'OCTETS (longueur: 1 à 4 octets)</p> <p>ABORT (P)</p> <p>Étiquette de type de message: 01100111</p> <p>Longueur de type de message: nombre correct d'octets</p> <p>Étiquette d'ID de transaction de destination: 01001001</p> <p>Longueur d'ID de transaction de destination: nombre correct d'octets</p> <p>Valeur d'ID de transaction de destination: CHAÎNE D'OCTETS (longueur: 1 à 4 octets) (valeur d'OTID reçue dans le message UNKNOWN)</p> <p>Étiquette de cause P-Abort: 01001010</p> <p>Longueur de cause P-Abort: nombre correct d'octets</p> <p>Valeur de cause P-Abort: type de message non reconnu 00000000</p>					

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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)</p> <p><i>Dem. TR-BEGIN</i></p> <p>=====></p> <p>BEGIN</p> <p>-----></p> <p><i>:Détection du type de message Unknown</i></p> <p><-----</p> <p>ABORT (P)</p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> </td> <td style="width: 40%; vertical-align: top; border: none;"> <p>-----></p> <p><-----</p> <p>-----></p> </td> <td style="width: 30%; vertical-align: top;"> <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>-----></p> <p><i>:Détection du type de message Unknown</i></p> <p><-----</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>-----></p> <p><i>:Détection du type de message Unknown</i></p> <p><-----</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>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.</p>					
DESCRIPTION DE L'ESSAI					
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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.2.7 3)	Feuillet: 2 de 2
DESCRIPTION DE L'ESSAI CONTINUED	
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>MESSAGE UNKNOWN</p> <p>Etiquette de type de message: unknown (par exemple, 01100110) 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 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>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 UNKNOWN)</p> <p>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</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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>:Détection de l'erreur de syntaxe</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)	:Détection de l'erreur de syntaxe	<-----	BEGIN	ABORT (P)	----->	
SP A (TSL)		SP B (TSL)								
:Détection de l'erreur de syntaxe	<-----	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 DE L'ESSAI										
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>Étiquette de type de message: 01100010 Longueur de type de message: nombre correct d'octets</p> <p>Étiquette non valide: par exemple, 00100010</p> <p>Étiquette 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>Étiquette de type de message: 01100111 Longueur de type de message: nombre correct d'octets</p> <p>Étiquette 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>Étiquette de cause P-Abort: 01001010 Longueur de cause P-Abort: nombre correct d'octets Valeur de cause P-Abort: partie transaction incorrecte 00000011</p>										

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.3.2 1)	Feillet: 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 D'ESSAI: 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><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>			
<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.</p>					
DESCRIPTION DE L'ESSAI					
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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.2.3.2 1)	Feuillet: 2 de 2
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>CONTINUE</p> <p>Etiquette de type de message: 01100101 Longueur de type de message: nombre correct d'octets</p> <p>Etiquette non valide: par exemple, 00011111</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 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>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 CONTINUE)</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>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.3.1 1)	Feuillet: 1 de 2										
RÉFÉRENCE: 3.3.4/Q.774											
TITRE: Messages incorporés; 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 supprimer 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 doit être à l'état IR/actif (TSL). 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 D'ESSAI: 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%; text-align: center;"><-----</td> <td style="width: 30%;">SP B (TSL)</td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">CONTINUE</td> </tr> <tr> <td>ABORT (P)</td> <td style="text-align: center;">-----></td> <td></td> </tr> </table>			SP A (TSL)	<-----	SP B (TSL)			CONTINUE	ABORT (P)	----->	
SP A (TSL)	<-----	SP B (TSL)									
		CONTINUE									
ABORT (P)	----->										
DESCRIPTION DE L'ESSAI											
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?										

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.3.1 1)	Feuillet: 2 de 2
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.3.2 1)	Feuillet: 1 de 1
RÉFÉRENCE: 3.3.4/Q.774	
TITRE: Messages incorporés; Type de message End	
SOUS-TITRE: Réception d'un message End à l'état de repos	
OBJET: Vérifier que, à la réception d'un message End avec un DTID non affecté, le point sémaphore A peut supprimer le message	
CONDITIONS INITIALES: Le point SP A (TSL) doit être à l'état de repos et le point SP B doit être à l'état IR/actif. Faire en sorte que les données au point SP B soient telles qu'un message End avec un DTID déductible mais non affecté soit envoyé au point SP A	
CONFIGURATION: 1	TYPE D'ESSAI: VAT
TYPE DE SP: SP	
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <p style="text-align: center;"> SP A (TSL) SP B (TSL) <----- END </p>	
DESCRIPTION DE L'ESSAI	
1.	Faire en sorte que le point SP B envoie le message End avec un DTID non affecté au point SP A.
2.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE END AU POINT SP A.
3.	VÉRIFICATION B: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE END.
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	
<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: nombre correct d'octets Valeur d'ID de transaction de destination: CHAÎNE D'OCTETS (longueur: 1 à 4 octets)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.3.3 1)	Feuillet: 1 de 1	
RÉFÉRENCE: 3.3.4/Q.774		
TITRE: Messages incorporés; Type de message Abort		
SOUS-TITRE: Réception d'un message Abort à l'état de repos		
OBJET: Vérifier que, à la réception d'un message Abort avec un DTID non affecté, le point sémaphore A peut supprimer le message		
CONDITIONS INITIALES: Le point SP A (TSL) doit être à l'état de repos et le point SP B doit être à l'état IR/actif. Faire en sorte que les données au point SP B soient telles qu'un message Abort avec un DTID déductible mais non affecté soit envoyé au point SP A		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="text-align: center;"> <p>SP A (TSL)</p> </div> <div style="text-align: center;"> <p><-----</p> </div> <div style="text-align: center;"> <p>SP B (TSL)</p> <p>ABORT (P)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Faire en sorte que le point SP B envoie le message Abort avec un DTID non affecté au point SP A.	
2.	VÉRIFICATION A: VÉRIFIER QUE L'UTILISATEUR-TR N'A PAS ÉTÉ INFORMÉ DU MESSAGE ABORT AU POINT SP A.	
3.	VÉRIFICATION B: VÉRIFIER QU'AUCUN MESSAGE N'A ÉTÉ CRÉÉ PAR LE POINT SP A EN RÉPONSE AU MESSAGE ABORT.	
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		
<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)</p> <p>Etiquette de cause P-Abort: 01001010 Longueur de cause P-Abort: un octet Valeur de cause P-Abort: NOMBRE ENTIER {0, 1, 2, 3, 4}</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.1 1)	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 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 D'ESSAI: VAT		
TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; 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><i>(Terminaison de base)</i></p> <p><i>(Terminer nouvelle transaction)</i></p> <p>END</p> <p><i>Ind. TR-END</i></p> <p><=====</p> </td> <td style="width: 50%; vertical-align: top; border-left: 1px dashed black;"> <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><i>(Terminaison de base)</i></p> <p><i>(Terminer nouvelle transaction)</i></p> <p>END</p> <p><i>Ind. TR-END</i></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><i>(Terminaison de base)</i></p> <p><i>(Terminer nouvelle transaction)</i></p> <p>END</p> <p><i>Ind. TR-END</i></p> <p><=====</p>	<p>SP B (TSL)</p> <p>BEGIN (nouvelle transaction)</p> <p>END</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.1 1)	Feuillet: 2 de 3
DESCRIPTION DE L'ESSAI	
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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.1 1)	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES	
<p>BEGIN (1^{er} message)</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) X</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>BEGIN (2^{ème} message)</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) Y</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>END (1^{er} message)</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: 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^{ème} message BEGIN)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>END (2^{ème} message)</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: 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)</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 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 D'ESSAI: VAT	TYPE DE SP: SP			
<p>SÉQUENCE DE MESSAGES ATTENDUE:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; vertical-align: top;"> <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>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: 10%; text-align: center; vertical-align: middle;"> <p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p> </td> <td style="width: 40%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</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-CONTINUE</i></p> <p><=====</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>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>CONTINUE</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-CONTINUE</i></p> <p><=====</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>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>BEGIN (nouvelle transaction)</p> <p>END</p>			

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.1 2)	Feuillet: 2 de 3
DESCRIPTION DE L'ESSAI	
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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.1 2)	Feuillet: 3 de 3
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 ^{ème} 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 ^{ème} message BEGIN)	
Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets	
END (2 ^{ème} 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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.2 1)	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 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 D'ESSAI: 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><-----</p> <p>ABORT (P)</p> <p>-----></p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p> </td> <td style="width: 35%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE (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>-----></p> <p><-----</p> <p>ABORT (P)</p> <p>-----></p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p>	<p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>CONTINUE (nouvelle transaction)</p> <p>END</p>
<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>-----></p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p>	<p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>CONTINUE (nouvelle transaction)</p> <p>END</p>			

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.2 1)	Feuillet: 2 de 3
DESCRIPTION DE L'ESSAI	
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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.2 1)	Feuillet: 3 de 3
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: 01001001B 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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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><i>Ind. TR-CONTINUE</i></p> <p><=====</p> <p>ABORT (P)</p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p> </td> <td style="width: 30%; vertical-align: top; text-align: center;"> <p>-----></p> <p><-----</p> <p><-----</p> <p>-----></p> <p><-----</p> </td> <td style="width: 35%; vertical-align: top;"> <p>SP B (TSL)</p> <p>CONTINUE</p> <p>CONTINUE (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-CONTINUE</i></p> <p><=====</p> <p>ABORT (P)</p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p>	<p>-----></p> <p><-----</p> <p><-----</p> <p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>CONTINUE (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-CONTINUE</i></p> <p><=====</p> <p>ABORT (P)</p> <p><i>Ind. TR-P-ABORT</i></p> <p><=====</p> <p><i>Ind. TR-END</i></p> <p><=====</p>	<p>-----></p> <p><-----</p> <p><-----</p> <p>-----></p> <p><-----</p>	<p>SP B (TSL)</p> <p>CONTINUE</p> <p>CONTINUE (nouvelle transaction)</p> <p>END</p>			

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.2.2)	Feuillet: 2 de 3
DESCRIPTION DE L'ESSAI	
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 en réponse au message Begin émanant du point SP A.
3.	Faire en sorte que le point SP B envoie un message Continue avec un DTID non affecté au point SP A.
4.	Faire en sorte que le point SP B réponde par un message End au message Begin.
5.	VÉRIFICATION A: LE MESSAGE BEGIN A-T-IL ÉTÉ ENVOYÉ CORRECTEMENT PAR LE POINT SP A?
6.	VÉRIFICATION B: LES MESSAGES CONTINUE ONT-ILS ÉTÉ REÇUS CORRECTEMENT PAR LE POINT SP A?
7.	VÉRIFICATION C: LE DTID DANS LE MESSAGE ABORT ÉTAIT-IL IDENTIQUE À L'OTID DANS LE DEUXIÈME MESSAGE CONTINUE?
8.	VÉRIFICATION D: LA CAUSE P-ABORT DANS LE MESSAGE ABORT ÉTAIT-ELLE LA VALEUR CORRECTE (ID DE TRANSACTION NON RECONNU)?
9.	VÉRIFICATION E: LE MESSAGE END A-T-IL ÉTÉ REÇU CORRECTEMENT PAR LE POINT SP A?
10.	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	
<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>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 1.4.2.2)	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES ÉLÉMENTS D'INFORMATION CONTENUS DANS LES MESSAGES	
CONTINUE (2 ^{ème} 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: 01001001B 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 ^{ème} 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	

Remplacée par une version plus récente

7.2 Spécification d'essais de la sous-couche composant TC

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

- a) Pour tous les essais, l'expression «... composant avec information correcte» dans la description de l'essai signifie que les valeurs détaillées dans le composant indiqué seront syntaxiquement vérifiées en fonction des informations qui figurent dans la table de vérification des composants contenus dans les messages.
- b) Dans certains essais, il est nécessaire de vérifier que l'automate de lancement d'opération est revenu à l'état de repos. On peut, à cet effet, envoyer un composant Return Result-Last (Résultat complet) avec l'identificateur Invoke ID (Identificateur de lancement) supposé au repos. Si l'IUT (Instance sous test) renvoie un composant Reject (Rejet) avec code de problème Invoke ID non reconnu, cela signifie que le résultat de cette vérification est positif.
- c) Pour tous les essais de la sous-couche composant, le composant doit être transporté dans un message TSL; par exemple, le composant Invoke dans l'essai n° 2.1.1.1 est acheminé du point SP A au point SP B dans un message Begin (Etablissement de transaction) et le composant Return Result-Last (Résultat complet) est acheminé dans un message End (Fin). En fait, si une transaction est initialement établie entre les points SP A et SP B, il est possible de transporter les composants Invoke et Return Result dans des messages Continue (Continuation).
- d) L'hypothèse sur laquelle se fondent ces essais CSL est que la transaction est maintenue jusqu'à ce que le dernier composant du flux de messages ait été transmis à l'entité homologue. Dans le cas où cette hypothèse n'est pas valable pour une application donnée (par exemple, à cause de l'utilisation d'un message Abort ou End), aucune appréciation concluante ne peut être portée sur l'essai.
- e) Les essais CSL supposent que la sous-couche transaction et le SCCP (sous-système commande de connexion sémaphore) fonctionnent correctement. Ils supposent ainsi, en particulier, que les composants sont acheminés dans des messages TSL valides et dans des états de transaction valides afin que des occurrences anormales n'interviennent pas dans la sous-couche ou les sous-couches de base.
- f) Les informations relatives à l'Utilisateur-TC, telles que le code d'opération et les paramètres spécifiques, ne sont pas spécifiés. Il incombe aux personnes chargées de l'exécution des essais d'inclure, s'il y a lieu, les informations en fonction de l'application donnée, pour provoquer le flux de composants prévu.

7.2.2 Liste d'essais de la sous-couche composant

Tous les essais sont des essais de validation.

Les essais marqués d'un astérisque «*» sont des essais de compatibilité.

2 *Sous-couche composant*

2.1 Fonctions valides

2.1.1 Composant Invoke, opérations non corrélées

2.1.1.1 Lancement 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 erreur
 - * 2.1.1.1.4 IUT récepteur: notification erreur
 - * 2.1.1.1.5 IUT émetteur: expiration temporisateur
- ##### 2.1.1.2 Lancement d'opération simple de classe 2
- * 2.1.1.2.1 IUT émetteur: réception erreur
 - * 2.1.1.2.2 IUT émetteur: expiration temporisateur
- ##### 2.1.1.3 Lancement 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 Lancement d'opération simple de classe 4
- * 2.1.1.4.1 IUT émetteur

Remplacée par une version plus récente

- 2.1.2 Composant Invoke, opérations corrélées
 - 2.1.2.1 Lancement d'opération initiale de classe 1
 - * 2.1.2.1.1 IUT émetteur: réception d'un lancement d'opération corrélée de classe 1, notification résultat
 - * 2.1.2.1.2 IUT récepteur: émission d'un lancement d'opération corrélée de classe 1, réception résultat
 - * 2.1.2.1.3 IUT émetteur: réception d'un lancement d'opération corrélée de classe 1, notification erreur
 - * 2.1.2.1.4 IUT récepteur: envoi d'un lancement d'opération corrélée de classe 1, réception erreur
 - 2.1.2.2 Lancement d'opération initiale de classe 4
 - * 2.1.2.2.1 IUT émetteur: réception d'une demande de lancement d'opération corrélée de classe 2, pas de résultat
 - * 2.1.2.2.2 IUT récepteur: envoi d'une demande de lancement d'opération corrélée de classe 2, expiration 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 (Lancement)
 - 2.1.3.1.3 Code de problème Return Result (Retour Résultat)
 - 2.1.3.1.4 Code de problème Return Error (Retour Erreur)
 - 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 Lancement d'opération de classe 1
 - 2.1.3.3.2 Lancement d'opération de classe 2
 - 2.1.3.3.3 Lancement d'opération de classe 3
 - 2.1.3.3.4 Lancement 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 corrélée non prévue
 - 2.1.4.1.3 Réponse corrélée 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 Lancement d'opération simple de classe 1
 - * 2.1.5.1.1 IUT émetteur: réception composants segmentés
 - * 2.1.5.1.2 IUT récepteur: émission composants segmentés
 - 2.1.5.2 Lancement d'opération simple de classe 3
 - * 2.1.5.2.1 IUT émetteur: réception 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 Invoke ID (Identificateur de lancement)
 - 2.1.7.4.1.1 Invoke ID = -127(FFh)
 - 2.1.7.4.1.2 Invoke ID = 0(00h)
 - 2.1.7.4.2 Code d'opération global
- 2.1.8 Groupement de composants multiples
 - 2.1.8.1 Lancement d'opérations multiples: réception succès
 - 2.1.8.2 Lancement d'opérations multiples: notification succès
 - 2.1.8.3 Composant mal formé reçu

Remplacée par une version plus récente

- 2.2 Comportement syntaxiquement non valide
 - 2.2.1 Valeurs non valides des éléments d'information
 - 2.2.1.1 Longueur de l'Identificateur Invoke ID supérieure à 1 dans le composant Invoke
 - 2.2.1.2 Longueur de l'Identificateur Invoke ID 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 Etiquette 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 (inconnu)
 - 2.2.2.4.1 Identificateur Invoke ID non reconnaissable
 - 2.2.2.4.2 Identificateur Invoke ID déductible
 - 2.2.3 Codage non valide du composant Invoke
 - 2.2.3.1 Etiquette non valide
 - 2.2.3.2 Longueur de composant erronée
 - 2.2.3.3 EOC (indicateur de fin de données) manquant dans la forme indéfinie
- 2.3 Comportement inopportun
 - 2.3.1 Composant Invoke inopportun
 - 2.3.1.1 ID corrélé non valide
 - 2.3.2 ID Invoke non reconnu
 - 2.3.2.1 Composant Return Result-Last (Résultat complet) inopportun
 - 2.3.2.2 Composant Return Result Not-Last (Résultat partiel) 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 la classe 2
 - 2.3.3.2 Return Result-Last pour la classe 4
 - 2.3.3.3 Return Result Not-Last pour la classe 2
 - 2.3.3.4 Return Result Not-Last pour la classe 4
 - 2.3.3.5 Return Error pour la classe 3
 - 2.3.3.6 Return Error pour la classe 4

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.1	Feuillet: 1 de 2	
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées		
SOUS-TITRE: Lancement d'opération simple de classe 1; IUT émetteur: réception résultat		
OBJET: Vérifier qu'une opération simple de classe 1 peut être lancée avec succès et que l'indication d'exécution avec succès de l'opération peut être reçue et transmise à l'utilisateur-TC		
<p>CONDITIONS INITIALES:</p> <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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <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> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p><-----</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: i (i représente un nombre entier)		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>COMPOSANT RETURN RESULT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de Composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.2	Feuille: 1 de 2	
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées		
SOUS-TITRE: Lancement d'opération simple de classe 1; IUT récepteur: notification résultat		
OBJET: Vérifier qu'une opération de classe 1 peut être lancée 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU:</p> <pre style="font-family: monospace;"> SP A (CSL) SP B (CSL) <----- INVOKE (i) TC-INVOKE Ind. <===== TC-RESULT-L Dem. =====> RETURN-RESULT-LAST (i) -----> </pre>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.3	Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774	
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées	
SOUS-TITRE: Lancement d'opération simple de classe 1; IUT émetteur: réception erreur	
OBJET: Vérifier qu'une opération de classe 1 peut être lancée 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <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> </div> <div style="width: 10%; text-align: center;"> <p>-----></p> <p><-----</p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>RETURN ERROR (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.3	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.4	Feillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774	
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées	
SOUS-TITRE: Lancement d'opération simple de classe 1; IUT récepteur: notification d'erreur	
OBJET: Vérifier qu'une opération de classe 1 peut être lancée avec succès et que l'indication de non exécution avec succès de l'opération peut être envoyée	
<p>CONDITIONS INITIALES:</p> <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 de l'utilisateur-TC au point SP A soient telles qu'un composant Return Error puisse être créé</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT et CPT
TYPE DE SP: SP	
<p>FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Ind. TC-INVOKE</i> <=====</p> <p><i>Dem. TC-U-ERROR</i> =====></p> <p>RETURN-ERROR (i) -----></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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	
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant Invoke dans un message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.4	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de Composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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</p> <p>paramètres (fournis par l'utilisateur-TC)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.1.5	Feuille: 1 de 1
RÉFÉRENCE: 3.2.1/Q.774	
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées	
SOUS-TITRE: Lancement d'opération simple de classe 1; IUT émetteur: expiration temporisateur	
OBJET: Vérifier qu'une opération de classe 1 peut être lancée 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i). -----></p> <p>expiration du temporisateur pour le lancement (i)</p> <p><i>Ind. TC-L-CANCEL</i> =====></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.2.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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</p> <p>paramètres (fournis par l'utilisateur-TC)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.2.2	Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1/Q.774	
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées	
SOUS-TITRE: Lancement d'opération simple de classe 2; IUT émetteur: expiration temporisateur	
OBJET: Vérifier qu'une opération de classe 2 peut être lancée 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) -----></p> <p>expiration temporisateur pour le lancement (i)</p> <p><i>Ind. TC-L-CANCEL</i></p> <p><=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement d'une opération de classe 2 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'UTILISATEUR-TC AU POINT SP A A-T-IL ÉTÉ INFORMÉ DE L'EXPIRATION DU TEMPORISATEUR?
5.	VÉRIFICATION D: L'AUTOMATE DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.3.1	Feillet: 1 de 2	
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées		
SOUS-TITRE: Lancement d'opération simple de classe 3; IUT émetteur: réception résultat		
OBJET: Vérifier qu'une opération simple de classe 3 peut être lancée avec succès et que l'indication d'exécution avec succès de l'opération peut être reçue et transmise à l'utilisateur-TC		
<p>CONDITIONS INITIALES:</p> <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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <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> </div> <div style="width: 10%; text-align: center;"> <p>-----></p> <p><-----</p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération simple de classe 3 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES		
<p>Partie Composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100</p> <p>Longueur de partie composant: nombre correct d'octets</p> <p>Composant Invoke dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE)</p> <p>Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010</p> <p>Longueur d'ID de lancement: 00000001 (un octet)</p> <p>ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.3.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100011 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.3.2	Feuillet: 1 de 1
RÉFÉRENCE: 3.2.1/Q.774	
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées	
SOUS-TITRE: Lancement d'opération simple de classe 3; IUT émetteur: expiration temporisateur	
OBJET: Vérifier qu'une opération de classe 3 peut être lancée avec succès et que l'indication d'expiration de temporisateur peut être transmise à l'utilisateur-TC	
<p>CONDITIONS INITIALES:</p> <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'aucun composant ne puisse être créé</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT et CPT
TYPE DE SP: SP	
<p>FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU:</p> <p style="text-align: center;">SP A (CSL) SP B (CSL)</p> <p style="text-align: center;"><i>Dem. TC-INVOKE</i></p> <p style="text-align: center;">=====></p> <p style="text-align: center;">INVOKE (i) -----></p> <p style="text-align: center;">expiration du temporisateur pour le lancement (i)</p> <p style="text-align: center;"><i>Ind. TC-L-CANCEL</i></p> <p style="text-align: center;"><=====</p>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES	
<p>Partie Composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.1.4.1	Feillet: 1 de 1
RÉFÉRENCE: 3.2.1/Q.774	
TITRE: Fonctions valides; Composant Invoke, opérations non corrélées	
SOUS-TITRE: Lancement 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
<p>FLUX DE MESSAGE ET DE COMPOSANTS ATTENDU:</p> <p style="margin-left: 40px;">SP A (CSL) SP B (CSL)</p> <p style="margin-left: 40px;"><i>Dem. TC-INVOKE.</i></p> <p style="margin-left: 40px;">=====></p> <p style="margin-left: 40px;">INVOKE (i) -----></p> <p style="margin-left: 40px;">expiration du temporisateur pour le lancement (i)</p> <p style="margin-left: 40px;"><i>Ind. TC-L-CANCEL</i></p> <p style="margin-left: 40px;"><=====</p>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?
TABLE DE VÉRIFICATION DES ÉLÉMENTS CONTENUS DANS LES MESSAGES	
<p>Partie composant dans les messages TSL</p> <p style="margin-left: 40px;">Etiquette de partie composant: 01101100</p> <p style="margin-left: 40px;">Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p style="margin-left: 40px;">Etiquette de type de composant: 10100001 (INVOKE)</p> <p style="margin-left: 40px;">Longueur de composant: nombre correct d'octets</p> <p style="margin-left: 40px;">Etiquette d'ID de lancement: 00000010</p> <p style="margin-left: 40px;">Longueur d'ID de lancement: 00000001 (un octet)</p> <p style="margin-left: 40px;">ID de lancement: i (i représente un nombre entier)</p> <p style="margin-left: 40px;">Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale)</p> <p style="margin-left: 40px;">Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si x est long d'un octet)</p> <p style="margin-left: 40px;">Code d'opération: x (x représente un code d'opération valide)</p> <p style="margin-left: 40px;">paramètres (fournis par l'utilisateur-TC)</p>	

Remplacée par une version plus récente

NUMERO DE L'ESSAI: 2.1.2.1.1	Feuillet: 1 de 3			
RÉFÉRENCE: 3.2.1/Q.774				
TITRE: Fonctions valides; Composant Invoke, opérations corrélées				
SOUS-TITRE: Lancement d'opération initiale de classe 1; IUT émetteur; réception d'un lancement d'opération corrélée de classe 1; notification résultat.				
OBJET: Vérifier qu'une opération corrélée de classe 1 peut être reçue avec succès et que l'indication d'exécution avec succès de l'opération initiale peut être reçue correctement.				
<p>CONDITIONS INITIALES:</p> <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 Invoke corrélé puisse être créé</p>				
CONFIGURATION: 1	TYPE D'ESSAI: VAT et CPT			
TYPE DE SP: SP				
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; vertical-align: top;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i)</p> <p><i>Ind. TC-INVOKE</i> <=====</p> <p><i>Dem. TC-RESULT-L</i> =====></p> <p>RETURN-RESULT-LAST (j)</p> <p><i>Ind. TC-RESULT-L</i> <=====</p> </td> <td style="width: 40%; border: none; text-align: center; vertical-align: middle;"> <p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (CSL)</p> <p>INVOKE (j, i)</p> <p>RETURN-RESULT-LAST (i)</p> </td> </tr> </table>		<p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i)</p> <p><i>Ind. TC-INVOKE</i> <=====</p> <p><i>Dem. TC-RESULT-L</i> =====></p> <p>RETURN-RESULT-LAST (j)</p> <p><i>Ind. TC-RESULT-L</i> <=====</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>
<p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i)</p> <p><i>Ind. TC-INVOKE</i> <=====</p> <p><i>Dem. TC-RESULT-L</i> =====></p> <p>RETURN-RESULT-LAST (j)</p> <p><i>Ind. TC-RESULT-L</i> <=====</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 DE L'ESSAI				
1.	Lancement d'une opération corrélée 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 CORRÉLÉ 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?			

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.1	Feuillet: 2 de 3
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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)	
Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.1	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL envoyé du point SP B au point SP A</p> <p>Etiquette de type de Composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.2	Feuillet: 1 de 3	
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant-Invoke, opérations corrélées		
SOUS-TITRE: Lancement d'opération initiale de classe 1; IUT récepteur; émission d'un lancement d'opération corrélée de classe 1; réception résultat		
OBJET: Vérifier qu'une opération corrélée de classe 1 peut être lancée 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 lancera une opération corrélée		
CONFIGURATION: 1	TYPE D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre style="font-family: monospace; font-size: 1.2em;"> 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 DE L'ESSAI		
1.	Lancement d'une opération corrélée 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 CORRÉLÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?
4.	VÉRIFICATION C:	L'ID CORRÉLÉ 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.2	Feuille: 2 de 3
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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)	
Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.2	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL envoyé par le point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.3	Feuillet: 1 de 3	
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations corrélées		
SOUS-TITRE: Lancement d'opération initiale de classe 1; IUT émetteur; réception d'un lancement d'opération corrélée de classe 1; notification erreur		
OBJET: Vérifier qu'une opération corrélée 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 corrélée puisse être créée		
CONFIGURATION: 1	TYPE D'ESSAI: VAT et CPT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANT ATTENDU:		
SP A (CSL) <i>Dem. TC-INVOKE</i> =====>	-----> <----- -----> <-----	SP B (CSL) INVOKE (j, i) RETURN-RESULT-LAST (i)
<i>Ind. TC-INVOKE</i> <=====		
<i>Dem. TC-U-ERROR</i> =====>		
RETURN-ERROR (j) <i>Ind. TC-RESULT-L</i> <=====		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération corrélée 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 CORRÉLÉ 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.3	Feuillet: 2 de 3
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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 les messages TSL envoyés par le point SP B	
Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets	
Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)	
Etiquette d'ID corrélée: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: 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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.3	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL envoyé par le point SP B</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.4	Feuillet: 1 de 3	
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations corrélées		
SOUS-TITRE: Lancement d'opération initiale de classe 1; IUT récepteur: envoi d'un lancement d'opération corrélée de classe 1; réception erreur		
OBJET: Vérifier qu'une opération corrélée de classe 1 peut être lancée 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="text-align: center;"><i>Ind. TC-INVOKE</i> <=====</p> <p style="text-align: center;"><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (j, i) -----></p> <p style="text-align: center;"><i>Ind. TC-U-ERROR</i> <=====</p> <p style="text-align: center;"><i>Dem. TC-RESULT-L</i> =====></p> <p>RETURN-RESULT-LAST (i) -----></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> <p style="text-align: center;">-----<</p> <p>RETURN ERROR (j)</p> <p style="text-align: center;">-----<</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération corrélée 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 CORRÉLÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?
4.	VÉRIFICATION C:	L'ID CORRÉLÉ 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.4	Feuillet: 2 de 3
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 les messages TSL du point SPB au point SP A	
Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets	
Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 les messages TSL envoyés par le point SP	
Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets	
Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)	
Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.1.4	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans les messages TSL envoyés par le point SP A</p> <p>Etiquette de type de Composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

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

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.2.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant INVOKE dans le message TSL envoyé par le point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)</p> <p>Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: i</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.2.2	Feuillet: 1 de 2	
RÉFÉRENCE: 3.2.1/Q.774		
TITRE: Fonctions valides; Composant Invoke, opérations corrélées		
SOUS-TITRE: Lancement d'opération initiale de classe 4; IUT récepteur; envoi d'une demande de lancement d'opération corrélée de classe 2; expiration temporisateur		
OBJET: Vérifier qu'une opération corrélée de classe 2 peut être lancée avec succès et que l'opération initiale de classe 4 peut être exécutée avec succès		
CONDITIONS INITIALES: Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne un composant Invoke qui lancera une opération corrélée de classe 2		
CONFIGURATION: 1	TYPE D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 20px;"><i>Ind. TC-INVOKE</i> ←=====</p> <p style="margin-left: 20px;"><i>Dem. TC-INVOKE</i> =====→</p> <p>INVOKE (j, i) expiration du temporisateur pour le lancement (j)</p> <p style="margin-left: 20px;"><i>Ind. TC-L-CANCEL</i></p> </div> <div style="width: 10%; text-align: center;"> <p>←-----</p> <p>-----→</p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération corrélée 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 CORRÉLÉ AVEC INFORMATION CORRECTE A-T-IL ÉTÉ ENVOYÉ PAR LE POINT SP A?
4.	VÉRIFICATION C:	L'ID CORRÉLÉ ÉTAIT-IL IDENTIQUE À L'IDENTIFICATEUR INVOKE ID INITIAL ENVOYÉ PAR LE POINT SP B?
5.	VÉRIFICATION D:	L'AUTOMATE DE LANCEMENT 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: i (i représente un nombre entier)		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.2.2.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant INVOKE dans le message TSL envoyé du point SPA au point SP B</p> <p>Etiquette de type de composant: 10100010 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)</p> <p>Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: i</p> <p>Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) Longueur de code d'opération: nombre correct d'octets (par exemple, 00000001 si y représente un code d'opération valide) Code d'opération: y (y représente un code d'opération valide)</p> <p>paramètres (fournis par l'utilisateur-TC)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.1.1	Feuillet: 1 de 2	
RÉFÉRENCES: 3.2.1/Q.774, 3.8.1/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		
<p>CONDITIONS INITIALES:</p> <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 Reject puisse être créé avec code de problème général</p>		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 30%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i)</p> <p><i>Ind. TC-R-REJECT</i></p> <p><=====</p> </div> <div style="width: 30%; text-align: center;"> <p>-----></p> <p><-----</p> </div> <div style="width: 30%;"> <p>SP B (CSL)</p> <p>REJECT (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: i (i représente un nombre entier)		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.1.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000000 (problème général) Longueur de code de problème: 00000001 Code de problème: 00000000 (composant non reconnu)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.1.2	Feuillet: 1 de 2	
RÉFÉRENCES: 3.2.1/Q.774; 3.8.2/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:		
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 Invoke puisse être créé		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:		
SP A (CSL) <i>Ind. TC-INVOKE</i> =====> <i>Dem. TC-INVOKE</i> =====> INVOKE (j, i) -----> <i>Ind. TC-R-REJECT</i> <=====	<----- <----- -----> <-----	SP B (CSL) INVOKE (i) REJECT (j)
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération corrélée 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 DE LANCEMENT 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: i (i représente un nombre entier)		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.1.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)</p> <p>Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: i</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j</p> <p>Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 (un octet) Code de problème: 00000101 (ID corrélé non reconnu)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.1.3	Feuillet: 1 de 2	
RÉFÉRENCE: 3.2.1/Q.774, 3.8.3/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 lancée 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 D'ESSAI: 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) REJECT (i)
<i>Dem. TC-RESULT-L</i> =====>		
RETURN RESULT-LAST (i)		
<i>Ind. TC-R-REJECT</i> <=====		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: i (i représente un nombre entier)		

Remplacée par une version plus récente

NUMERO DE L'ESSAI: 2.1.3.1.3	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème: 00000001 Code de problème: 00000000 (ID de lancement non reconnu)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.1.4	Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774, 3.8.4/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 lancée avec succès et que le rejet distant peut être reçu et transmis à l'utilisateur-TC	
<p>CONDITIONS INITIALES:</p> <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 Reject puisse être créé</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Ind. TC-INVOKE</i> <=====</p> <p><i>Dem. TC-U-ERROR</i> =====></p> <p>RETURN ERROR (i)</p> <p><i>Ind. TC-R-REJECT</i> <=====</p> </div> <div style="width: 10%; text-align: center;"> <p><-----</p> <p>-----></p> <p><-----</p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> <p>REJECT (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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	
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.1.4	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant : nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000001 (ID de lancement non reconnu)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.2.1	Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774, 3.8.2/Q.772	
TITRE: Fonctions valides; Rejet distant	
SOUS-TITRE: Rejet distant par l'utilisateur-TC; Code de problème Invoke C	
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 avec code de problème Invoke puisse être créé	
CONFIGURATION: 1	TYPE D'ESSAI: 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>REJECT (i)</p>
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
Partie composant dans les messages 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 de lancement: 00000010	
Longueur d'ID de lancement: 00000001 (un octet)	
ID de lancement: i (i représente un entier)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.2.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000000 (ID de lancement dupliqué)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.2.2	Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774, 3.8.3/Q.772	
TITRE: Fonctions valides; Rejet distant	
SOUS-TITRE: Rejet distant par l'utilisateur-TC; Code de problème Return Result	
OBJET: Vérifier qu'un rejet distant par l'utilisateur-TC, avec code de problème Return Result, peut être reçu et transmis à l'utilisateur-TC	
<p>CONDITIONS INITIALES:</p> <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 Reject avec code de problème Return Result puisse être créé</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Ind. TC-INVOKE</i> <=====</p> <p><i>Dem. TC-RESULT-L</i> =====></p> <p>RETURN RESULT-LAST (i) -----></p> <p><i>Ind. TC-U-REJECT</i> <=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> <p>REJECT (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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	
<p>Partie composant dans les messages de TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.2.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de Composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème : 00000001 Code de problème: 00000010 (paramètre type erroné)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.2.3	Feuillet: 1 de 2
RÉFÉRENCE: 3.2.1/Q.774, 3.8.4/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	
<p>CONDITIONS INITIALES:</p> <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 Return Error puisse être créé</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-INVOKE</i> <=====</p> <p style="margin-left: 40px;"><i>Dem. TC-U-ERROR</i> =====></p> <p>RETURN ERROR (i)</p> <p style="margin-left: 40px;"><i>Ind. TC-U-REJECT</i> <=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> <p style="margin-left: 40px;">-----></p> <p>REJECT (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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	
<p>Partie composant dans les messages de TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message de TSL du point SP B au SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un entier)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.2.3	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000010 (erreur non reconnue)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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; Lancement d'opération de classe 1		
OBJET: Vérifier qu'une opération simple de classe 1 peut être lancée 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 D'ESSAI: 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>INVOKE (i) -----></p> <p><----- REJECT (i)</p> <p><i>Ind. TC-U-REJECT</i> <=====</p>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: i (i représente un entier)		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.3.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000010 (paramètre type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.3.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000010 (paramètre type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.3.3	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000010 (paramètre type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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; Lancement d'opération de classe 4	
OBJET: Vérifier qu'une opération simple de classe 4 peut être lancée 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 D'ESSAI: VAT
TYPE DE SP: SP	
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-U-REJECT</i> <=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p><----- REJECT (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un entier)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.3.3.4	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000010 (paramètre type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.1.1	Feuillet: 1 de 2	
RÉFÉRENCE: 3.2.2.2/Q.774		
TITRE: Fonctions valides; Réception d'un composant entraînant un rejet par l'utilisateur-TC		
SOUS-TITRE: Problème Invoke; Code d'opération non reconnu		
OBJET: Vérifier qu'un rejet d'une opération demandée peut être effectué		
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-INVOKE</i></p> <p style="margin-left: 40px;"><=====</p> <p style="margin-left: 40px;"><i>Dem. TC-U-REJECT</i></p> <p style="margin-left: 40px;">=====></p> <p>REJECT (i)</p> </div> <div style="width: 10%; text-align: center;"> <p><-----</p> <p>-----></p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération du point SP B au point SP A avec un code d'opération non reconnu.	
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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.1.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000001 (type de problème INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000001 (opération non reconnue)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 corrélée non prévue	
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause d'une opération corrélée non prévue, sans que la demande de lancement 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 corrélé soit contenu dans un message TSL approprié	
CONFIGURATION: 1	TYPE D'ESSAI: VAT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC INVOKE</i> <=====</p> <p><i>Dem. TC-U-REJECT</i> =====></p> <p>REJECT (j) -----></p> <p><i>Ind. TC-RESULT-L</i> <=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (j, i)</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement d'une opération non corrélée 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 CORRÉLÉ 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.1.2	Feuillet: 2 de 3
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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un entier)	
Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: 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 corrélé 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j	
Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000111 (opération corrélée non prévue)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.1.2	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
Composant RETURN RESULT-LAST dans les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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.	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 corrélée non prévue		
OBJET: Vérifier qu'une réponse corrélée non prévue peut être rejetée		
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 qui lancera une opération corrélée 2) Faire en sorte que les données au point SP B soient telles qu'une réponse corrélée contienne au moins un paramètre non associé au résultat de l'opération		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC INVOKE</i> <=====</p> <p><i>Dem. TC-U-REJECT</i> =====></p> <p>REJECT (j) -----></p> <p><i>Ind. TC-RESULT-L</i> <=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (j, i)</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 CORRÉLÉ 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.1.3	Feuille: 2 de 3
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 A au point SP B	
Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets	
Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un 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 corrélée)	
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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)	
Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: 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: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.1.3	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000111 (réponse corrélée non prévue)</p> <p>Composant RETURN RESULT-LAST dans le message TSL envoyé par le point SP B</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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) Código de operación: x (voir la Note)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.1.4	Feuillet: 1 de 2	
RÉFÉRENCE: 3.2.2.2/Q.774		
TITRE: Fonctions valides; Réception d'un composant entraînant un rejet par l'utilisateur-TC		
SOUS-TITRE: Problème Invoke; Paramètre type erroné		
OBJET: Vérifier qu'un rejet d'une opération demandée peut être effectué		
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Ind. TC-INVOKE</i> <=====</p> <p><i>Dem. TC-U-REJECT</i> =====></p> <p>REJECT (i) -----></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération du point SP B au point SP A avec un paramètre type erroné inclus.	
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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.1.4	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC, y compris au moins un paramètre non associé à l'opération)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000001 (type de problème INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000010 (paramètre type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.2.1	Feuillet: 2 de 2
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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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: y (y est différent de x) (voir la Note)	
paramètres (fournis par l'utilisateur-TC)	
Composant REJECT dans les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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.	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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:		
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 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		
CONFIGURATION: 1	TYPE D'ESSAI: 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-U-ERROR</i> <=====		
<i>Dem. TC-U-REJECT</i> =====>		
REJECT (i)	----->	
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.3.1	Feuillet: 2 de 2
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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i	
Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale) Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si x 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i	
Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000010 (erreur non reconnue)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.3.2	Feuillet 2 de 2
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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement : 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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 lancée)	
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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i	
Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000011 (erreur non prévue)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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		
<p>CONDITIONS INITIALES:</p> <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 avec un paramètre type erroné soit créé</p>		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-U-ERROR</i> <=====</p> <p><i>Dem. TC-U-REJECT</i> =====></p> <p>REJECT (i) -----></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>RETURN ERROR (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.4.3.3	Feuillet: 2 de 2
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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i	
Etiquette de code d'erreur: 00000010 (locale) ou 00000110 (globale) Longueur de code d'erreur: nombre correct d'octets (par exemple, 00000001 si x 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 étiquette de paramètre non incluse dans les étiquettes 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i	
Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000100 (paramètre type erroné)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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: Lancement d'opération simple de classe 1; IUT émetteur; réception 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 D'ESSAI: VAT et CPT												
TYPE DE SP: SP													
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; vertical-align: top;"> SP A (CSL) <i>Dem. TC-INVOKE</i> =====> </td> <td style="width: 40%; vertical-align: top; text-align: center;"> -----> <----- <----- </td> <td style="width: 30%; vertical-align: top;"> SP B (CSL) RETURN RESULT NOT-LAST (i) RETURN RESULT-LAST (i) </td> </tr> <tr> <td style="vertical-align: top;"> INVOKE (i) </td> <td></td> <td></td> </tr> <tr> <td style="vertical-align: top;"> <i>Ind. TC-RESULT-NL</i> <===== </td> <td></td> <td></td> </tr> <tr> <td style="vertical-align: top;"> <i>Ind. TC-RESULT-L</i> <===== </td> <td></td> <td></td> </tr> </table>		SP A (CSL) <i>Dem. TC-INVOKE</i> =====>	-----> <----- <-----	SP B (CSL) RETURN RESULT NOT-LAST (i) RETURN RESULT-LAST (i)	INVOKE (i)			<i>Ind. TC-RESULT-NL</i> <=====			<i>Ind. TC-RESULT-L</i> <=====		
SP A (CSL) <i>Dem. TC-INVOKE</i> =====>	-----> <----- <-----	SP B (CSL) RETURN RESULT NOT-LAST (i) RETURN RESULT-LAST (i)											
INVOKE (i)													
<i>Ind. TC-RESULT-NL</i> <=====													
<i>Ind. TC-RESULT-L</i> <=====													
DESCRIPTION DE L'ESSAI													
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)													

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.5.1.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Etiquette de code d'opération: 00000010 (locale) ou 00000110 (globale) Longueur de code d'opération: nombre correct d'octets (para exemple, 00000001 si x est long d'un octet) Code d'opération: x (x représente un code d'opération valide)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT NOT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT NOT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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: Lancement 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	
<p>CONDITIONS INITIALES:</p> <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) Provoquer une stimulation de l'utilisateur-TC telle qu'un composant Return Result Not-last puisse être créé</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT et CPT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <p style="text-align: center;">SP A (CSL) SP B (CSL)</p> <p><i>Ind. TC-INVOKE</i> </p> <p style="text-align: center;"> INVOKE (i)</p> <p>RETURN RESULT NOT-LAST (i) </p> <p><i>Dem. TC-RESULT-NL</i> </p> <p>RETURN RESULT LAST (i) </p> <p><i>Dem. TC-RESULT-L</i> </p>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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	
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.5.1.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT NOT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>Paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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: Lancement d'opération simple de classe 3; IUT émetteur: réception 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	
<p>CONDITIONS INITIALES:</p> <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 Not-Last puisse être créé</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT et CPT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-RESULT-NL</i> <=====</p> <p><i>Ind. TC-RESULT-L</i> <=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p><----- RETURN RESULT NOT-LAST (i)</p> <p><----- RETURN RESULT-LAST (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.5.2.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT NOT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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'un lancement d'opération peut être annulé par 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 D'ESSAI: VAT et CPT
TYPE DE SP: SP	
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Dem. TC-U-CANCEL</i> =====></p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (i) -----></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.6	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème: 00000001 Code de problème: 00000000 (ID de lancement non reconnu)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.1	Feuillet: 1 de 2
RÉFÉRENCE: 3.3/Q.773	
TITRE: Fonctions valides; Variations de codage	
SOUS-TITRE: Longueur de composant: courte définie	
OBJET: Vérifier qu'une partie composant avec une forme courte 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 D'ESSAI: VAT
TYPE DE SP: SP	
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-INVOKE</i></p> <p style="margin-left: 40px;"><=====</p> <p style="margin-left: 40px;"><i>Dem. TC-RESULT-L</i></p> <p style="margin-left: 40px;">=====></p> <p style="margin-left: 40px;">RETURN RESULT-LAST (i)</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> <p>-----></p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 courte définie) Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100011 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT
TYPE DE SP: SP	
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-INVOKE</i></p> <p style="margin-left: 40px;"><=====</p> <p style="margin-left: 40px;"><i>Dem. TC-RESULT-L</i></p> <p style="margin-left: 40px;">=====></p> <p style="margin-left: 40px;">RETURN RESULT-LAST (i) -----></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00000010 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.3	Feuillet: 1 de 2
RÉFÉRENCE: 3.3/Q.773	
TITRE: Fonctions valides; Variations de codage	
SOUS-TITRE: Longueur de composant: indéfinie	
OBJET: Vérifier qu'une partie composant avec une forme indéfinie 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 D'ESSAI: VAT
TYPE DE SP: SP	
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-INVOKE</i></p> <p style="margin-left: 40px;"><=====</p> <p style="margin-left: 40px;"><i>Dem. TC-RESULT-L</i></p> <p style="margin-left: 40px;">=====></p> <p style="margin-left: 40px;">RETURN RESULT-LAST (i)</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> <p>-----></p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 indéfinie) Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.3	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Etiquette EOC: 0000 0000 Longueur EOC: 0000 0000</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.4.1.1	Feillet: 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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-INVOKE</i> <=====</p> <p style="margin-left: 40px;"><i>Dem. TC-RESULT-L</i> =====></p> <p>RETURN RESULT-LAST (i) -----></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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		
<p>Partie composant dans les messages TSL</p> <p style="margin-left: 40px;">Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p style="margin-left: 40px;">Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p style="margin-left: 40px;">Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 11111111 (FFh)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.4.1.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 11111111 (FFh)</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.4.1.2	Feillet: 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 D'ESSAI: VAT et CPT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-INVOKE</i> ←=====</p> <p style="margin-left: 40px;"><i>Dem. TC-RESULT-L</i> =====→</p> <p>RETURN RESULT-LAST (i) -----→</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: 0		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.4.1.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 0</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre style="font-family: monospace;"> SP A (CSL) SP B (CSL) <-----> INVOKE (i) Ind. TC-INVOKE <===== <===== Dem. TC-U-REJECT =====> REJECT (i) -----> </pre>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.7.4.2	Feuillet: 2 de 2
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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i	
Etiquette de code de problème: 10000001 (Type de problème INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000001 (opération non reconnue)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.8.1	Feillet: 1 de 2	
RÉFÉRENCE: Q.774		
TITRE: Fonctions valides; Groupement de composants multiples		
SOUS-TITRE: Lancement d'opérations multiples: réception succès		
OBJET: Vérifier que des opérations multiples peuvent être lancées avec succès et que les indications positives d'exécution de ces opérations peuvent être reçues correctement		
<p>CONDITIONS INITIALES:</p> <p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP A contienne des composants multiples</p> <p>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</p>		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE (#1)</i> =====></p> <ul style="list-style-type: none"> • • • <p><i>Dem. TC-INVOKE (#n)</i> =====></p> </div> <div style="width: 45%; text-align: center;"> <p>SP B (CSL)</p> <p>INVOKE #1, ..., #n^{a)} -----></p> <p>RETURN RESULT-LAST #1, ..., #n^{a)} -----<</p> </div> </div> <div style="margin-top: 20px;"> <p><i>Ind. TC-RESULT-L (#1)</i> -----<</p> <ul style="list-style-type: none"> • • • <p><i>Ind. TC-RESULT-L (#n)</i> =====></p> </div> <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 DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT (1, ..., n) ÉTAIENT-ILS AU REPOS AU POINT SP A?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.8.1	Feuille: 2 de 2
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	
Composants 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 x1 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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.8.2	Feuillet: 1 de 2
RÉFÉRENCE: Q.774	
TITRE: Fonctions valides; Groupement de composants multiples	
SOUS-TITRE: Lancement d'opérations multiples: notification succès	
OBJET: Vérifier que des opérations multiples peuvent être lancées avec succès et que les indications positives d'exécution de ces opérations peuvent être envoyées	
<p>CONDITIONS INITIALES:</p> <p>1) Provoquer une stimulation de l'utilisateur-TC telle qu'un message TSL approprié créé au point SP B contienne des composants multiples</p> <p>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</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> SP A (CSL) SP B (CSL) </div> <div style="text-align: center; margin-top: 10px;"> <p>INVOKE #1, ..., #n^{a)}</p> <p>←-----</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="margin-top: 20px;">RETURN RESULT-LAST #n, ..., #1^{a)}</p> <p>-----→</p> </div> <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 DE L'ESSAI	
1.	Lancement 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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.8.2	Feuillet: 2 de 2
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	
Composants 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 supprimés 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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 30%;"> <p>SP A (CSL)</p> <p><i>Ind. TC-INVOKE (#1)</i> <=====</p> <p><i>Ind. TC-L-REJECT (#2)</i> <=====</p> <p><i>Dem. TC-RESULT-L (#1)</i> =====></p> </div> <div style="width: 40%; text-align: center;"> <p>INVOKE #1, #2, #3 (Note 1)</p> <p><-----</p> <p>(#2 mal structuré, par exemple code d'opération manquant)</p> </div> <div style="width: 30%;"> <p>SP B (CSL)</p> <p>REJECT #2, RETURN RESULT-LAST #1</p> <p>-----></p> <p>(Note 2)</p> </div> </div>		
<p>NOTES</p> <ol style="list-style-type: none"> 1 La séquence des composants Invoke est importante 2 La séquence de ces composants n'est pas importante 		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100</p> <p>Longueur de partie composant: nombre correct d'octets</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.8.3	Feuillet: 2 de 3
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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.1.8.3	Feillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
Composant RETURN RESULT-LAST #1 dans les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 1	
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	
paramètres (fournis par l'utilisateur-TC)	
Composant REJECT #2 dans les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 2	
Etiquette de code de problème: 10000000 (problème général) Longueur de code de problème: 00000001 Code de problème: 00000010 (composant mal structuré)	
NOTE – Omis lorsque aucun paramètre n'est présent	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.1.1	Feuillet: 1 de 2	
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 supérieure à 1 dans le composant Invoke		
OBJET: Vérifier qu'un rejet d'une opération demandée peut être effectué à cause du codage incorrect de l'ID de composant (valeur en dehors des limites spécifiées)		
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-L-REJECT</i></p> <p style="margin-left: 40px;"><=====</p> <p>REJECT (NULL)</p> </div> <div style="width: 10%; text-align: center;"> <p><-----</p> <p>-----></p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération du point SP B au point SP A avec un identificateur Invoke ID égal à 2 octets (valeur interdite).	
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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000010 (deux octets) ID de lancement: 129</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.1.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette NULL: 00000101 Longueur NULL: 00000000</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.1.2	Feuillet: 1 de 2	
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-L-REJECT</i></p> <p style="margin-left: 40px;"><=====</p> <p>REJECT (NULL)</p> </div> <div style="width: 10%; text-align: center;"> <p><-----</p> <p>-----></p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>INVOKE</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération du point SP B au point SP A avec un identificateur Invoke ID égal à 0 octet (valeur interdite).	
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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000000 (zéro octet)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.1.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette NULL: 00000101 Longueur NULL: 00000000</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.1.1	Feuillet: 1 de 2	
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (NULL)</p> </div> <div style="width: 10%; text-align: center;"> <p><-----</p> <p>-----></p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>INVOKE</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.1.1	Feuillet: 2 de 2
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: 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)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.1.2	Feuillet: 1 de 2	
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p style="margin-left: 40px;"><i>Ind. TC-L-REJECT</i></p> <p style="margin-left: 40px;"><=====</p> <p>REJECT (i)</p> </div> <div style="width: 10%; text-align: center;"> <p><-----</p> <p>-----></p> </div> <div style="width: 45%;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p> <p>paramètres (fournis par l'utilisateur-TC)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.1.2	Feuillet: 2 de 2
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: 10100100 (REJECT) Longueur de composant: nombre correct d'octets	
Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i	
Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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		
<p>CONDITIONS INITIALES:</p> <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 sans identificateur Invoke ID puisse être créé</p>		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <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 le lancement (i)</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>RETURN RESULT-LAST</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.2.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette NULL: 00000101 Longueur NULL: 00000000</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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	
<p>CONDITIONS INITIALES:</p> <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 D'ESSAI: VAT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <p style="text-align: center;">SP A (CSL) SP B (CSL)</p> <p style="text-align: center;"><i>Dem. TC-INVOKE</i> =====></p> <p style="text-align: center;">INVOKE (i) -----></p> <p style="text-align: center;"><----- RETURN RESULT-LAST (i)</p> <p style="text-align: center;"><i>Ind. TC-L-REJECT</i> <=====</p> <p style="text-align: center;">REJECT (i) -----></p>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.2.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 Longueur de séquence: nombre correct d'octets</p> <p>paramètres (doivent être fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.2.3		Feillet: 1 de 2
RÉFÉRENCE: 6.4/Q.773; 3.2.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Composant Return Result; Etiquette de séquence manquante mais paramètres inclus		
OBJET: Vérifier qu'un rejet peut être initialisé avec succès à cause de l'absence d'étiquette de séquence, bien que les paramètres soient inclus		
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 message TSL approprié contienne un composant Return Result-Last avec une étiquette de séquence non valide		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (i) -----></p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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'AUTOMATE DE LANCEMENT 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: i (i représente un nombre entier)		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.2.3	Feuille: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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)</p> <p>paramètres (doivent être fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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		
<p>CONDITIONS INITIALES:</p> <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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <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 le lancement (i)</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>RETURN ERROR</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.3.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>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 lancée)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette NULL: 00000101 Longueur NULL: 00000000</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: 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>INVOKE (i) -----></p> <p><----- RETURN ERROR (i)</p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (i) -----></p>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010		
Longueur d'ID de lancement: 00000001 (un octet)		
ID de lancement: i (i représente un nombre entier)		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.3.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000101 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000001 (composant type erroné)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.4.1	Feillet: 1 de 1	
RÉFÉRENCE: 3.2.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Type de composant Unknown; Identificateur Invoke ID non reconnaissable		
OBJET: Vérifier qu'un rejet peut être initialisé à cause d'un type de composant Unknown avec identificateur Invoke ID non reconnu		
CONDITIONS INITIALES: Provoquer une stimulation telle qu'un message TSL approprié créé au point SP B contienne un composant Unknown comme indiqué ci-dessous		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="text-align: center;"> <p>SP A (CSL)</p> <p><i>Ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (NULL)</p> </div> <div style="text-align: center; flex-grow: 1;"> <p><-----</p> <p>-----></p> </div> <div style="text-align: center;"> <p>SP B (CSL)</p> <p>Composant Unknown</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération du point SP B au point SP A avec un type de composant Unknown ayant un quelconque contenu.	
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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant Unknown dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: n'importe quelle valeur sauf 10100001, 10100010, 10100011, 10100100 et 10100111 Longueur de composant: nombre correct d'octets Contenu du composant: quelconque</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette NULL: 00000101 Longueur NULL: 00000000</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000000 (composant non reconnu)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.4.2	Feuillet: 1 de 2	
RÉFÉRENCE: 3.2.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Structure non valide		
SOUS-TITRE: Type de composant Unknown; Identificateur Invoke ID déductible		
OBJET: Vérifier qu'un rejet peut être initialisé à cause d'un type de composant Unknown avec identificateur Invoke ID déductible		
CONDITIONS INITIALES: Provoquer une stimulation telle qu'un message TSL approprié créé au point SP B contienne un composant Unknown avec identificateur Invoke ID déductible comme indiqué ci-dessous		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="text-align: center;"> <p>SP A (CSL)</p> <p><i>Ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (i ou NULL)</p> </div> <div style="text-align: center; flex-grow: 1;"> <p><-----</p> <p>-----></p> </div> <div style="text-align: center;"> <p>SP B (CSL)</p> <p>Composant Unknown (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération du point SP B au point SP A avec un type de composant Unknown comme indiqué ci-dessous.	
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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant Unknown dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: n'importe quelle valeur sauf 10100001, 10100010, 10100011, 10100100 et 10100111 Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.2.4.2	Feuillet: 2 de 2
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: 10100100 (REJECT) Longueur de composant: nombre correct d'octets	
Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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: 00000000 (composant non reconnu)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.3.1	Feuillet: 1 de 2	
RÉFÉRENCE: 3.2.2.2/Q.774; 3.2.3/Q.773		
TITRE: Comportement syntaxiquement non valide; Codage non valide du composant Invoke		
SOUS-TITRE: Etiquette non valide		
OBJET: Vérifier qu'un rejet est déclenché à cause d'une étiquette non valide		
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre style="margin-left: 40px;"> SP A (CSL) SP B (CSL) <----- INVOKE (i) <===== Ind. TC-L-REJECT <===== REJECT (i ou NULL) -----></pre>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération du point SP B au point SP A avec étiquette non valide.	
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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.3.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Etiquette non valide: 00011111 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)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>ou</p> <p>Etiquette NULL: 00000101 Longueur NULL: 00000000</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000010 (composant mal structuré)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.3.2	Feuillet: 1 de 2	
RÉFÉRENCE: 3.2.2.2/Q.774		
TITRE: Comportement syntaxiquement non valide; Codage non valide du composant Invoke		
SOUS-TITRE: Longueur de composant erronée		
OBJET: Vérifier qu'un rejet d'opération demandée peut être initialisé à cause d'une longueur de composant erronée		
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 30%;"> <p>SP A (CSL)</p> <p><i>Ind. TC-L-REJECT</i></p> <p><=====</p> <p>REJECT (i ou NULL)</p> </div> <div style="width: 30%; text-align: center;"> <p><-----</p> <p>-----></p> </div> <div style="width: 30%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement d'une opération du point SP B au point SP A avec une valeur de longueur de composant non valide.	
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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre erroné d'octets (par exemple, 00000000)</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.3.2	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>ou</p> <p>Etiquette NULL: 00000101 Longueur NULL: 00000000</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000010 (composant mal structuré)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.3.3	Feuillet: 1 de 2	
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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <pre style="margin-left: 40px;"> SP A (CSL) SP B (CSL) <----- INVOKE (i) <===== Ind. TC-L-REJECT <===== REJECT (i ou NULL) -----> </pre>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100 Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets (forme indéfinie)</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.2.3.3	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>ou</p> <p>Etiquette NULL: 00000101 Longueur NULL: 00000000</p> <p>Etiquette de code de problème: 10000000 (type de problème général) Longueur de code de problème: 00000001 Code de problème: 00000010 (composant mal structuré)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.1.1	Feuillet: 1 de 2
RÉFÉRENCE: 3.2.2/Q.774	
TITRE: Comportement inopportun; Composant Invoke inopportun	
SOUS-TITRE: ID corrélé non valide	
OBJET: Vérifier qu'un rejet d'opération demandée peut être initialisé à cause d'un ID corrélé non valide	
<p>CONDITIONS INITIALES:</p> <p>1) Provoquer une stimulation 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 Invoke corrélé peut être créé comme indiqué ci-dessous</p>	
CONFIGURATION: 1	TYPE D'ESSAI: VAT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (j) -----></p> <p>Expiration du temporisateur pour le lancement (i)</p> <p><i>Ind. TC-L-CANCEL</i> <=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>INVOKE (j, k)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100</p> <p>Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans les messages TSL du point SP A au point SP B</p> <p>Etiquette d'ID de lancement: 00000010</p> <p>Longueur d'ID de lancement: 00000001 (un octet)</p> <p>ID de lancement: i (i représente un nombre entier)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.1.1	Feillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant INVOKE dans le message TSL envoyé par le point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: j (j représente un nombre entier)</p> <p>Etiquette d'ID corrélé: 10000000 Longueur d'ID corrélé: 00000001 (un octet) ID corrélé: k (k est un nombre entier différent de i)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL envoyé par le point SP A</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j</p> <p>Etiquette de code de problème: 10000001 (INVOKE) Longueur de code de problème: 00000001 Code de problème: 00000101 (ID corrélé non reconnu)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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	
<p>CONDITIONS INITIALES:</p> <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 D'ESSAI: VAT
TYPE DE SP: SP	
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <div style="width: 45%;"> <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 le lancement (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> </div> <div style="width: 45%; border-left: 1px dashed black; padding-left: 10px;"> <p>SP B (CSL)</p> <p>RETURN RESULT-LAST (j)</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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?

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.2.1

Feuillet: 2 de 2

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 les messages 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 de lancement: 00000010
Longueur d'ID de lancement: 00000001 (un octet)
ID de lancement: 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 les messages 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 de lancement: 00000010
Longueur d'ID de lancement: 00000001
ID de lancement: 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 les messages 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 de lancement: 00000010
Longueur d'ID de lancement: 00000001
ID de lancement: 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.

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.2.2	Feuillet: 1 de 3	
RÉFÉRENCE: 3.2.2/Q.774		
TITRE: Comportement inopportun; ID Invoke non reconnu		
SOUS-TITRE: Composant Return Result Not-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 Not-Last reçu		
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 3 2) Faire en sorte que les données au point SP B soient telles qu'un composant Return Result Not-Last avec identificateur Invoke ID non valide puisse être créé		
CONFIGURATION: 1	TYPE D'ESSAI: 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 (j)
<i>Ind. TC-L-REJECT</i> <=====		
REJECT (j) Expiration du temporisateur pour le lancement (i)	----->	
<i>Ind. TC-L-CANCEL</i> <=====		
	<-----	RETURN RESULT NOT-LAST (i)
<i>Ind. TC-L-REJECT</i> <=====		
REJECT (i)	----->	
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 Not-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?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.2.2	Feuillet: 2 de 3
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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.2.2	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème: 00000001 Code de problème: 00000000 (identificateur Invoke ID non reconnu)</p> <p>Composant RETURN RESULT NOT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème: 00000001 Code de problème: 00000000 (identificateur Invoke ID non reconnu)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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					
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 non reconnue 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					
CONFIGURATION: 1	TYPE D'ESSAI: 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%; vertical-align: top;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (j) -----> Expiration du temporisateur pour le lancement (i)</p> <p><i>Ind. TC-L-CANCEL</i> <=====</p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (i) -----></p> </td> <td style="width: 40%; vertical-align: top; text-align: center;"> <p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p> <p>-----></p> </td> <td style="width: 30%; vertical-align: top;"> <p>SP B (CSL)</p> <p>RETURN ERROR (j)</p> <p>RETURN ERROR (i)</p> </td> </tr> </table>			<p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (j) -----> Expiration du temporisateur pour le lancement (i)</p> <p><i>Ind. TC-L-CANCEL</i> <=====</p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (i) -----></p>	<p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (CSL)</p> <p>RETURN ERROR (j)</p> <p>RETURN ERROR (i)</p>
<p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i> =====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (j) -----> Expiration du temporisateur pour le lancement (i)</p> <p><i>Ind. TC-L-CANCEL</i> <=====</p> <p><i>Ind. TC-L-REJECT</i> <=====</p> <p>REJECT (i) -----></p>	<p>-----></p> <p><-----</p> <p>-----></p> <p><-----</p> <p>-----></p>	<p>SP B (CSL)</p> <p>RETURN ERROR (j)</p> <p>RETURN ERROR (i)</p>			
DESCRIPTION DE L'ESSAI					
1.	Lancement 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?				

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.2.3	Feuillet: 2 de 3
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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: j	
Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000000 (identificateur Invoke ID non reconnu)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.2.3	Feuillet: 3 de 3
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>Composant RETURN ERROR dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000000 (identificateur Invoke ID non reconnu)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.2.4	Feillet: 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 aucun lancement actif n'a pas d'effet sur un lancement actif		
<p>CONDITIONS INITIALES:</p> <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</p> <p>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</p>		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>SP A (CSL)</p> <p><i>Dem. TC-INVOKE</i></p> <p>=====></p> <p>INVOKE (i) -----></p> <p><i>Ind. TC-R-REJECT a)</i></p> <p><=====</p> <p><i>Ind. TC-L-RESULT</i></p> <p><=====</p> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p>REJECT (j)</p> <p>-----<</p> <p>RETURN RESULT-LAST (i)</p> <p>-----<</p> </div> </div> <p>a) L'émission de l'indication TC-R-REJECT dépend du mode de mise en œuvre utilisé.</p>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.2.4	Feuillet: 2 de 2
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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: 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 B au point SP A	
Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets	
Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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 les messages 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: 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.	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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		
<p>CONDITIONS INITIALES:</p> <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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <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> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p><-----</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100</p> <p>Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type composant: 10100001 (INVOKE)</p> <p>Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010</p> <p>Longueur d'ID de lancement: 00000001 (un octet)</p> <p>ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.3.1	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème: 00000001 Code de problème: 00000001 (return result imprévu)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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		
<p>CONDITIONS INITIALES:</p> <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 D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <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> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p><-----</p> <p>RETURN RESULT-LAST (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100</p> <p>Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE)</p> <p>Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010</p> <p>Longueur d'ID de lancement: 00000001 (un octet)</p> <p>ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.3.2	Feuille: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100010 (RETURN RESULT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème: 00000001 Code de problème: 00000001 (return result imprévu)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT										
TYPE DE SP: SP											
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; vertical-align: top;"> SP A (CSL) <i>Dem. TC-INVOKE</i> =====> </td> <td style="width: 50%; vertical-align: top;"> SP B (CSL) </td> </tr> <tr> <td style="vertical-align: top;"> INVOKE (i) </td> <td style="vertical-align: middle; text-align: center;"> -----> </td> </tr> <tr> <td style="vertical-align: middle;"> <----- </td> <td style="vertical-align: top;"> RETURN RESULT NOT-LAST (i) </td> </tr> <tr> <td style="vertical-align: top;"> <i>Ind. TC-L-REJECT</i> <===== </td> <td></td> </tr> <tr> <td style="vertical-align: top;"> REJECT (i) </td> <td style="vertical-align: middle; text-align: center;"> -----> </td> </tr> </table>		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)	----->
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 DE L'ESSAI											
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)											

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.3.3	Feuille: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT NOT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème: 00000001 Code de problème: 00000001 (return result imprévu)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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		
<p>CONDITIONS INITIALES:</p> <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-Not-Last puisse être créé</p>		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <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> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p><-----</p> <p>RETURN RESULT NOT-LAST (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100</p> <p>Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE)</p> <p>Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010</p> <p>Longueur d'ID de lancement: 00000001 (un octet)</p> <p>ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.3.4	Feuille: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN RESULT NOT-LAST dans les messages TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100111 (RETURN RESULT NOT-LAST) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de séquence: 00110000 (voir la Note) Longueur de séquence: nombre correct d'octets (voir la Note)</p> <p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans les messages TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000010 (RETURN RESULT) Longueur de code de problème: 00000001 Code de problème: 00000001 (return result imprévu)</p> <p>NOTE – Omis lorsque aucun paramètre n'est présent.</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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		
<p>CONDITIONS INITIALES:</p> <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 Error puisse être créé</p>		
CONFIGURATION: 1	TYPE D'ESSAI: VAT	TYPE DE SP: SP
<p>FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU:</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; padding: 10px;"> <div style="width: 45%;"> <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> </div> <div style="width: 45%; text-align: right;"> <p>SP B (CSL)</p> <p><-----</p> <p>RETURN ERROR (i)</p> </div> </div>		
DESCRIPTION DE L'ESSAI		
1.	Lancement 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 DE LANCEMENT ÉTAIT-IL AU REPOS AU POINT SP A?	
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES		
<p>Partie composant dans les messages TSL</p> <p>Etiquette de partie composant: 01101100</p> <p>Longueur de partie composant: nombre correct d'octets</p> <p>Composant INVOKE dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100001 (INVOKE)</p> <p>Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010</p> <p>Longueur d'ID de lancement: 00000001 (un octet)</p> <p>ID de lancement: i (i représente un nombre entier)</p>		

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.3.5	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000001 (return error imprévu)</p>	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 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 D'ESSAI: VAT
TYPE DE SP: SP	
FLUX DE MESSAGES ET DE COMPOSANTS ATTENDU: <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p style="text-align: center;">SP A (CSL)</p> <p style="text-align: center;"><i>Dem. TC-INVOKE</i></p> <p style="text-align: center;">=====></p> <p style="text-align: center;">INVOKE (i)</p> <p style="text-align: center;"> </p> <p style="text-align: center;"><i>Ind. TC-L-REJECT</i></p> <p style="text-align: center;"><=====</p> <p style="text-align: center;">REJECT (i)</p> </div> <div style="width: 10%; text-align: center;"> <p>-----></p> <p><-----</p> <p>-----></p> </div> <div style="width: 45%;"> <p style="text-align: center;">SP B (CSL)</p> <p style="text-align: center;">RETURN ERROR (i)</p> </div> </div>	
DESCRIPTION DE L'ESSAI	
1.	Lancement 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 DE LANCEMENT 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 de lancement: 00000010 Longueur d'ID de lancement: 00000001 (un octet) ID de lancement: i (i représente un nombre entier)	

Remplacée par une version plus récente

NUMÉRO DE L'ESSAI: 2.3.3.6	Feuillet: 2 de 2
TABLE DE VÉRIFICATION DES COMPOSANTS CONTENUS DANS LES MESSAGES	
<p>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)</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant RETURN ERROR dans le message TSL du point SP B au point SP A</p> <p>Etiquette de type de composant: 10100011 (RETURN ERROR) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>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</p> <p>paramètres (fournis par l'utilisateur-TC)</p> <p>Composant REJECT dans le message TSL du point SP A au point SP B</p> <p>Etiquette de type de composant: 10100100 (REJECT) Longueur de composant: nombre correct d'octets</p> <p>Etiquette d'ID de lancement: 00000010 Longueur d'ID de lancement: 00000001 ID de lancement: i</p> <p>Etiquette de code de problème: 10000011 (RETURN ERROR) Longueur de code de problème: 00000001 Code de problème: 00000001 (return error imprévu)</p>	