

Groupe de travail Réseau
Request for Comments : 5546
 RFC rendue obsolète : 2446
 RFC mise à jour : 5545
 Catégorie : Sur la voie de la normalisation

C. Daboo, éd., Apple Inc.
 décembre 2009

Traduction Claude Brière de L'Isle

Protocole iCalendar d'inter fonctionnement indépendant du transport (iTIP)

Résumé

Le présent document spécifie un protocole qui utilise la spécification d'objet iCalendar pour fournir l'interopérabilité de programmation entre différents systèmes de calendrier. Cela est fait sans référence à un protocole de transport spécifique de façon à permettre plusieurs méthodes de communication entre systèmes. Des documents ultérieurs vont définir des profils de ce protocole qui utiliseront des méthodes spécifiques interopérables de communication entre systèmes.

Le protocole iCalendar d'interfonctionnement indépendant du transport (iTIP, *iCalendar Transport-Independent Interoperability Protocol*) complète la spécification d'objet iCalendar en ajoutant la sémantique des méthodes de programmation de groupes couramment disponibles dans les systèmes de calendrier actuels. Ces méthodes de programmation permettent à deux systèmes de calendrier ou plus d'effectuer des transactions comme la publication, la programmation, la reprogrammation, la réponse à des demandes de programmation, de négocier ou d'annuler des changements.

Statut de ce mémoire

Ceci est un document de l'Internet sur la voie de la normalisation.

Le présent document a été produit par l'équipe d'ingénierie de l'Internet (IETF). Il représente le consensus de la communauté de l'IETF. Il a subi une révision publique et sa publication a été approuvée par le groupe de pilotage de l'ingénierie de l'Internet (IESG). Tous les documents approuvés par l'IESG ne sont pas candidats à devenir une norme de l'Internet ; voir la Section 2 de la RFC5741.

Les informations sur le statut actuel du présent document, tout errata, et comment fournir des réactions sur lui peuvent être obtenues à <http://www.rfc-editor.org/info/rfc5546>

Notice de droits de reproduction

Copyright (c) 2009 IETF Trust et les personnes identifiées comme auteurs du document. Tous droits réservés.

Le présent document est soumis au BCP 78 et aux dispositions légales de l'IETF Trust qui se rapportent aux documents de l'IETF (<http://trustee.ietf.org/license-info>) en vigueur à la date de publication de ce document. Prière de revoir ces documents avec attention, car ils décrivent vos droits et obligations par rapport à ce document. Les composants de code extraits du présent document doivent inclure le texte de licence simplifié de BSD comme décrit au paragraphe 4.e des dispositions légales du Trust et sont fournis sans garantie comme décrit dans la licence de BSD simplifiée.

Le présent document peut contenir des matériaux provenant de documents de l'IETF ou de contributions à l'IETF publiées ou rendues disponibles au public avant le 10 novembre 2008. La ou les personnes qui ont le contrôle des droits de reproduction sur tout ou partie de ces matériaux peuvent n'avoir pas accordé à l'IETF Trust le droit de permettre des modifications de ces matériaux en dehors du processus de normalisation de l'IETF. Sans l'obtention d'une licence adéquate de la part de la ou des personnes qui ont le contrôle des droits de reproduction de ces matériaux, le présent document ne peut pas être modifié en dehors du processus de normalisation de l'IETF, et des travaux dérivés ne peuvent pas être créés en dehors du processus de normalisation de l'IETF, excepté pour le formater en vue de sa publication comme RFC ou pour le traduire dans une autre langue que l'anglais.

Table des matières

1. Introduction et vue d'ensemble.....	2
1.1 Conventions de formatage.....	2
1.2 Documents en rapport.....	3
1.3 Rôles.....	3
1.4 Méthodes.....	4
2. Modèles d'interopérabilité.....	4

2.1 Protocole d'application.....	5
3. Éléments de protocole d'application.....	7
3.1 Tableaux des restrictions communes aux composants.....	7
3.2 Méthodes pour les composants de calendrier VEVENT.....	8
3.3 Méthodes pour les composants VFREEBUSY.....	19
3.4 Méthodes pour les composants VTODO.....	21
3.5 Méthodes pour composants VJOURNAL.....	31
3.6 Réponses d'état.....	34
3.7 Considérations de mise en œuvre.....	38
4. Exemples.....	39
4.1 Exemples d'événement publié.....	39
4.2 Exemples d'événement de groupe.....	41
4.3 Exemples de temps occupés.....	49
4.4 Exemples d'événement récurrent et de zone horaire.....	50
4.5 Exemples de choses à faire de groupe.....	58
4.6 Exemples de journal.....	61
4.7 Autres exemples.....	62
5. Positions de repli de protocole d'application.....	64
5.1 Mise en œuvre partielle.....	64
5.2 Problèmes de latence.....	67
5.3 Numéro de séquence.....	67
6. Considérations sur la sécurité.....	67
6.1 Menaces sur la sécurité.....	67
6.2 Recommandations.....	68
6.3 Questions de confidentialité.....	69
7. Considérations relatives à l'IANA.....	69
7.1 Gabarit d'enregistrement pour les valeurs de REQUEST-STATUS.....	69
7.2 Ajouts au registre iCalendar METHOD.....	69
7.3 Registre des valeurs de REQUEST-STATUS.....	71
8. Remerciements.....	72
9. Références.....	72
9.1 Références normatives.....	72
9.2 Référence pour information.....	72
Appendice A. Différences avec la RFC 2446.....	72
A.1 Restrictions changées.....	72
A.2 Caractéristiques déconseillées.....	73
Adresse de l'éditeur.....	73

1. Introduction et vue d'ensemble

Le présent document spécifie comment les systèmes de calendrier utilisent les objets iCalendar [RFC5545] pour interopérer avec les autres systèmes de calendrier. En particulier, il spécifie comment programmer des événements, des choses à faire (*to-do*), ou des entrées de journal. Il spécifie de plus comment chercher les informations sur les heures d'occupation disponibles. Il fait cela de façon générale, sans spécifier comment a lieu en fait la communication entre les différents systèmes. Des documents futurs spécifieront les liens de transport entre les systèmes qui utilisent ce protocole.

Ce protocole se fonde sur des messages envoyés d'un émetteur à un ou plusieurs receveurs. Pour certains types de messages, un receveur peut répondre afin de mettre à jour son état et peut aussi retourner des informations d'état de transaction/demande. Le protocole prend en charge la capacité de l'émetteur du message de modifier ou annuler le message d'origine. Le protocole prend aussi en charge la capacité pour les receveurs de suggérer des changements à l'émetteur d'un message. Les éléments du protocole définissent aussi les rôles de l'utilisateur pour ses transactions.

La présente spécification rend obsolète la RFC 2446 – une liste des changements importants est fournie à l'Appendice A.

1.1 Conventions de formatage

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le

BCP 14, [RFC2119].

Les rôles de calendrier et de programmation sont mentionnés par des chaînes de texte entre guillemets avec le premier caractère de chaque mot en majuscule. Par exemple, "Organisateur" se réfère à un rôle de "Utilisateur de calendrier" (CU, *Calendar User*) dans le protocole de programmation.

Les composants de calendrier définis par la [RFC5545] sont mentionnés comme des chaînes de texte en majuscules entre guillemets. Tous les composants de calendrier commencent par la lettre "V". Par exemple, "VEVENT" se réfère à l'événement composant de calendrier, "VTODO" se réfère au composant de calendrier à faire, et "VJOURNAL" se réfère au composant de calendrier journal.

Les méthodes de programmation sont mentionnées avec des chaînes de texte en majuscules entre guillemets. Par exemple, "REQUEST" se réfère à la méthode pour demander qu'un composant de calendrier de programmation soit créé ou modifié ; "REPLY" se réfère à la méthode qu'utilise le receveur d'une demande pour mettre à jour son état avec l'"Organisateur" du composant de calendrier.

Les propriétés définies par la [RFC5545] sont mentionnées avec des chaînes de texte en majuscules entre guillemets, précédées du mot "propriété". Par exemple, la propriété "PARTICIPANT" se réfère à la propriété iCalendar utilisée pour porter l'adresse du calendrier d'un "Utilisateur de calendrier".

Les paramètres de propriétés définies par la présente spécification sont mentionnés par des chaînes de texte en majuscules entre guillemets, précédées par le terme "paramètre". Par exemple, le paramètre "VALUE" se réfère au paramètre de propriété iCalendar utilisé pour outrepasser le type de données par défaut pour une valeur de propriété.

Les valeurs énumérées définies par la présente spécification sont mentionnées avec un texte en majuscules, soit seul, soit précédé par le mot "valeur".

Dans les tableaux, le texte entre guillemets est spécifié sans les guillemets afin de minimiser la longueur du tableau.

1.2 Documents en rapport

Le lecteur aura besoin d'être familier de plusieurs autres spécifications qui, avec la présente, décrivent les normes Internet de calendrier et de programmation. Ces documents sont :

[RFC5545] - spécifie les objets, types de données, propriétés, et paramètres de propriété utilisés dans les protocoles, ainsi que les méthodes pour les représenter et les coder.

[RFC6047] - spécifie un lien de messagerie Internet pour iTIP.

La présente spécification ne tente pas de répéter les concepts ou définitions de ces autres spécifications. Lorsque possible, des références explicites sont faites aux autres spécifications.

1.3 Rôles

Les échanges des objets iCalendar pour les besoins des calendriers et programmation de groupes se produisent entre les "Utilisateurs de calendrier" (CU). Les CU prennent plusieurs rôles dans iTIP :

Rôle	Description
Organisateur	Le CU qui initie un échange prend le rôle de organisateur. Par exemple, le CU qui propose une réunion d'un groupe est l'organisateur.
Participant	Les CU qui sont inclus dans le message de programmation comme possibles receveurs de ce message de programmation. Par exemple, les CU auxquels l'organisateur demande de participer à la réunion d'un groupe prennent le rôle de participant.
Autre CU	Un CU qui n'est pas explicitement inclus dans un message de programmation, c'est-à-dire., ni l'organisateur ni un participant.

Noter que "ROLE" est aussi un paramètre descriptif de la propriété iCalendar "ATTENDEE". Il est utilisé pour porter le contexte descriptif d'un "participant" -- comme "président", "participant obligé", ou "participant non obligé" -- et n'a rien à voir avec le flux de travail de calendrier.

1.4 Méthodes

La liste des méthodes iTIP figure ci-dessous et leur usage et sémantique sont définies à la Section 3 de ce document.

Méthode	Description
PUBLISH	Utilisée pour publier un objet iCalendar chez un ou plusieurs "Utilisateurs de calendrier". Il n'y a pas d'interactivité entre le publieur et un autre "Utilisateur de calendrier". Un exemple pourrait inclure une équipe de base-ball publiant son programme.
REQUEST	Utilisée pour programmer un objet iCalendar avec d'autres "utilisateurs de calendrier". Les demandes sont interactives en ce qu'elles exigent que le receveur réponde en utilisant les méthodes de réponse. Des demandes de réunion, des demandes de temps d'occupation, l'allocation de tâches aux autres "Utilisateurs de calendrier" sont des exemples. Les REQUEST sont aussi utilisées par l'organisateur pour mettre à jour l'état d'un objet iCalendar.
REPLY	Une REPLY est utilisée en réponse à une demande de porter l'état de participant à l'organisateur. Les réponses sont couramment utilisées pour répondre aux demandes de réunion et de tâches.
ADD	Ajoute une ou plusieurs nouvelles instances à un objet iCalendar récurrent existant.
CANCEL	Annule une ou plusieurs instances d'un objet iCalendar existant.
REFRESH	Utilisé par un participant pour demander la dernière version d'un objet iCalendar.
COUNTER	Utilisé par un participant pour négocier un changement dans un objet iCalendar. Des exemples incluent la demande de changer une heure d'événement proposée ou changer la date prévue pour une tâche.
DECLINECOUNTER	Utilisé par l'organisateur pour refuser une contre proposition.

La programmation de groupe dans iTIP est accomplie en utilisant l'ensemble de méthodes de "demandes" et "réponses" décrites ci-dessus. La tableau suivant montre les méthodes utilisées par qui peut les envoyer.

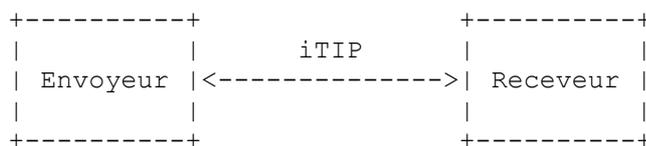
Générateur	Méthodes
organisateur	PUBLISH, REQUEST, ADD, CANCEL, DECLINECOUNTER
participant	REPLY, REFRESH, COUNTER, REQUEST (seulement en déléguant)

Noter que pour certains types de composants de calendrier, les méthodes admises sont un sous ensemble de celui ci-dessus. De plus, à part les composants iCalendar "VTIMEZONE", un seul type de composant est permis dans un message iTIP.

2. Modèles d'interopérabilité

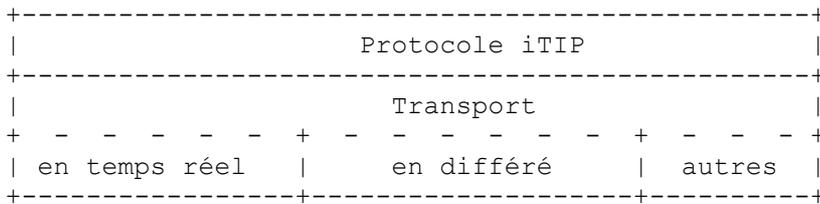
Il y a deux protocoles distincts pertinents pour l'interopérabilité : un "protocole d'application" et un "protocole de transport". Le protocole d'application définit le contenu des objets iCalendar envoyés entre envoyeur et receveur pour accomplir les transactions programmées mentionnées ci-dessus. Le protocole de transport définit comment les objets iCalendar sont envoyés entre l'envoyeur et le receveur. Le présent document se concentre sur le protocole d'application. Les documents de liaison comme la [RFC6047] se concentrent sur le protocole de transport.

La connexion entre envoyeur et receveur dans le diagramme ci-dessous se réfère au protocole d'application. Les objets iCalendar passés de l'envoyeur au receveur sont présentés à la Section 3, "Éléments de protocole d'application".



Il y a plusieurs variantes de ce diagramme dans lesquelles envoyeur et receveur prennent divers rôles de "Agent d'utilisateur de calendrier" (CUA) ou de "Service de calendrier" (CS).

L'architecture de iTIP est décrite dans le diagramme ci-dessous. Une application écrite selon la présente spécification peut fonctionner avec des liens pour le transport en différé, le transport en temps réel, ou les deux. Noter aussi que iTIP pourrait être lié à d'autres transports.



2.1 Protocole d'application

Dans le modèle iTIP, un objet iCalendar est créé et géré par un "Organisateur". L'"Organisateur" interagit avec les autres CU en envoyant un ou plusieurs des messages iTIP mentionnés ci-dessus. Les "Participants" utilisent la méthode "REPLY" pour communiquer leur état. Les "Participants" ne font pas de changements directs à l'objet iCalendar maître. Ils peuvent, cependant, utiliser la méthode "COUNTER" pour suggérer des changements à l'"Organisateur". Dans tous les cas, l'"Organisateur" a le contrôle complet de l'objet iCalendar maître.

2.1.1 État de programmation

Deux états distincts sont pertinents pour les objets iCalendar utilisés à la programmation : l'état global de l'objet iCalendar et l'état associé à un "Participant" à cet objet iCalendar.

L'état d'un objet iCalendar est défini par la propriété "STATUS" et est contrôlé par l'"Organisateur." Il n'y a pas de valeur par défaut pour la propriété "STATUS". L'"Organisateur" règle la propriété "STATUS" à la valeur appropriée pour chaque objet iCalendar.

L'état d'un "Participant" particulier à l'égard d'un objet iCalendar utilisé pour la programmation est défini par le paramètre "PARTSTAT" dans la propriété "ATTENDEE" pour chaque "Participant". Quand un "Organisateur" produit l'objet iCalendar initial, l'état du "participant" est normalement inconnu. L'"Organisateur" le spécifie en réglant le paramètre "PARTSTAT" à "NEEDS-ACTION". Chaque "Participant" modifie sa propriété "ATTENDEE" de paramètre "PARTSTAT" à une valeur appropriée au titre d'un message "REPLY" renvoyé à l'"Organisateur".

2.1.2 Délégation

La délégation est définie comme le processus par lequel un "Participant" accorde à un autre CU (ou plusieurs CU) le droit de participer en son nom. L'"Organisateur" est informé de ce changement parce que le "Participant" qui délègue en informe l'"Organisateur". Ces étapes sont détaillées dans les sections de méthode "REQUEST" pour les composants appropriés.

2.1.3 Action au nom d'autres utilisateurs de calendriers

Dans de nombreuses organisations, un utilisateur va agir au nom d'un autre pour organiser et/ou répondre aux demandes de réunions. iTIP fournit deux mécanismes qui prennent en charge ces activités.

D'abord, l'"Organisateur" est traité comme une entité spéciale, séparée des "Participants". Toutes les réponses des "Participants" s'écoulent vers l'"Organisateur", rendant facile de séparer un "Utilisateur de calendrier" qui organise une réunion des "utilisateurs de calendrier" qui participent à la réunion. De plus, iCalendar fournit des rôles descriptifs pour chaque "Participant". Par exemple, un rôle de "président" peut être prescrit à un ou plusieurs "Participants". Le "président" et l'"Organisateur" peuvent ou non être le même "Utilisateur de calendrier". Cela va bien dans les scénarios où un assistant peut gérer la logistique des réunions pour un autre individu qui préside la réunion.

Ensuite, un paramètre "SENT-BY" peut être spécifié dans les propriétés "Organisateur" ou "Participant". Quand il est spécifié, le paramètre "SENT-BY" indique que le CU qui répond agissait au nom du "Participant" ou "Organisateur" spécifié.

2.1.4 Révisions de composant

La propriété "SEQUENCE" est utilisée par l'"Organisateur" pour indiquer des révisions du composant de calendrier. Quand l'"Organisateur" fait des changements à une des propriétés suivantes, le numéro de séquence DOIT être incrémenté :

- o "DTSTART"
- o "DTEND"
- o "DURATION"
- o "DUE"

- o "RRULE"
- o "RDATE"
- o "EXDATE"
- o "STATUS"

De plus, les changements faits par l'"Organisateur" aux autres propriétés PEUVENT aussi exiger que le numéro de séquence soit incrémenté. Le CUA "Organisateur" DOIT incrémenter le numéro de séquence chaque fois qu'il fait des changements aux propriétés dans le composant de calendrier dont l'"Organisateur" estime qu'il vont faire disparaître la validité de l'état de participation des "Participants". Par exemple, changer la localisation d'une réunion d'un endroit à un autre distant pourrait effectivement impacter l'état de participation des "Participants".

Selon la "METHOD", la propriété "SEQUENCE" DOIT suivre ces règles dans le contexte de iTIP :

- o Pour les méthodes "PUBLISH" et "REQUEST", la valeur de propriété "SEQUENCE" est incrémentée en accord avec les règles déclarées ci-dessus.
- o La valeur de propriété "SEQUENCE" DOIT être incrémentée chaque fois que l'"Organisateur" utilise les méthodes "ADD" ou "CANCEL".
- o La valeur de propriété "SEQUENCE" NE DOIT PAS être incrémentée quand on utilise "REPLY", "REFRESH", "COUNTER", "DECLINECOUNTER", ou quand on envoie une demande de délégation.

Dans certaines circonstances, l'"Organisateur" peut ne pas avoir reçu de réponse à la révision finale envoyée. Dans cette situation, l'"Organisateur" peut souhaiter envoyer une "REQUEST" de mise à jour et régler "RSVP=TRUE" pour tous les "Participants" afin que les réponses courantes puissent être collectées.

La valeur de la propriété "SEQUENCE" contenue dans une réponse provenant d'un "Participant" ne peut pas toujours correspondre à la révision de l'"Organisateur". Les mises en œuvre peuvent choisir que le CUA indique au CU que la réponse est à un objet iCalendar qui a été révisé, et que le CU décide si il accepte ou non la réponse.

Bien qu'un changement du numéro de séquence indique un changement significatif d'un élément programmé antérieurement, les CUA de "Participant" NE DEVRAIENT PAS s'appuyer seulement sur un changement de numéro de séquence comme moyen de détecter un changement significatif. Les CUA DEVRAIENT plutôt comparer la version ancienne et la nouvelle des composants de calendrier, déterminer la nature exacte des changements, et prendre une décision – éventuellement fondée sur les préférences de l'"Utilisateur de calendrier" – sur si l'utilisateur devrait être explicitement informé du changement.

2.1.5 Séquençage des messages

Les CUA qui traitent le protocole d'application iTIP doivent souvent corrélérer un composant dans une mémorisation de calendriers avec un composant reçu dans le message iTIP. Par exemple, un événement peut être mis à jour avec une révision ultérieure du même événement. Pour accomplir cela, un CUA doit corrélérer la version de l'événement qui est déjà dans sa mémorisation de calendrier avec la version envoyée dans le message iTIP. En plus de cette corrélation, il y a plusieurs facteurs qui peuvent causer l'arrivée des messages iTIP dans un ordre inattendu. C'est-à-dire qu'un "Organisateur" pourrait recevoir une réponse à une révision antérieure d'un composant après la réception d'une réponse à une révision ultérieure.

Pour maximiser l'interopérabilité et traiter les messages qui arrivent dans un ordre inattendu, on utilise les règles suivantes :

1. La clé principale pour faire référence à un composant iCalendar particulier est la valeur de propriété "UID". Pour faire référence à une instance d'un composant récurrent, la clé principale est composée des propriétés "UID" et "RECURRENCE-ID".
2. La clé secondaire pour faire référence à un composant est la valeur de propriété "SEQUENCE". Pour les composants où les valeurs de propriété "UID" et "RECURRENCE-ID" sont les mêmes, le composant de plus haute valeur numérique pour la propriété "SEQUENCE" rend obsolètes toutes les autres révisions du composant qui ont des valeurs inférieures.
3. Les "Participants" envoient des messages "REPLY" à l'"Organisateur". Pour les réponses où les valeurs de propriété "UID" et "RECURRENCE-ID" sont les mêmes, la valeur de la propriété "SEQUENCE" indique la révision du composant à laquelle le "Participant" répond. La réponse avec la plus forte valeur numérique pour la propriété "SEQUENCE" rend obsolètes toutes les autres réponses avec des valeurs inférieures.
4. Dans les situations où les valeurs de propriété "UID", "RECURRENCE-ID", et "SEQUENCE" correspondent, la propriété "DTSTAMP" est utilisée pour le départage. Le composant qui a le "DTSTAMP" le plus récent outrepassé tous

les autres. De même, pour les réponses de "Participant" où les valeurs de propriété "UID", "RECURRENCE-ID", et "SEQUENCE" correspondantes, la réponse qui a le "DTSTAMP" le plus récent outrepassé toutes les autres.

Donc, les CUA ont besoin de faire persister les propriétés de composant suivantes afin de traiter correctement les messages iTIP : "UID", "RECURRENCE-ID", "SEQUENCE", et "DTSTAMP". De plus, pour chaque propriété "ATTENDEE" d'un composant, les CUA "organisateur" vont avoir besoin de faire persister les valeurs de propriété "SEQUENCE" et "DTSTAMP" associées à la dernière réponse du "Participant", afin que toutes les réponses antérieures venant d'un "Participant" qui sont reçues déclassées (par exemple, à cause de retards dans le transport) puissent être éliminées correctement.

3. Éléments de protocole d'application

Les messages iTIP sont des entités MIME "text/calendar" qui contiennent des informations de calendrier et de programmation. Le type particulier de message iCalendar est appelé le "type de méthode". Chaque type de méthode est identifié par une propriété "METHOD" spécifiée au titre du type de contenu "text/calendar". Le tableau ci-dessous montre les diverses combinaisons de composants de calendrier et les types de méthodes que prend en charge la présente spécification .

	VEVENT	VTODO	VJOURNAL	VFREEBUSY
PUBLISH	Oui	Oui	Oui	Oui
REQUEST	Oui	Oui	Non	Oui
REFRESH	Oui	Oui	Non	Non
CANCEL	Oui	Oui	Oui	Non
ADD	Oui	Oui	Oui	Non
REPLY	Oui	Oui	Non	Oui
COUNTER	Oui	Oui	Non	Non
DECLINECOUNTER	Oui	Oui	Non	Non

Chaque type de méthode est défini en termes de ses composants et propriétés associés. Certains composants et propriétés sont requis, certains sont facultatifs, et d'autres sont exclus. Les restrictions sont exprimées dans ce document en utilisant un simple "tableau des restrictions". La première colonne indique le nom d'un composant ou propriété. Les propriétés de l'objet iCalendar ne sont pas en retrait. Les propriétés d'un composant sont en retrait. La seconde colonne (la colonne "Présence") indique si un composant ou propriété devrait être présent ou non et, si il est présent, combien de fois il peut apparaître. La troisième colonne contient des commentaires pour des précisions complémentaires.

La colonne "Présence" utilise les valeurs suivantes pour indiquer si une propriété est exigée ou facultative, et le nombre de fois qu'elle peut apparaître dans l'objet iCalendar.

Valeur de présence Description

1	une instance DOIT être présente
1+	au moins une instance DOIT être présente
0	aucune instance de cette propriété NE DOIT être présente
0+	plusieurs instances PEUVENT être présentes
0 ou 1	au plus une instance de cette propriété PEUT être présente

Les tableaux mentionnent aussi des lignes "IANA-PROPERTY", "X-PROPERTY", "IANA-COMPONENT", et "X-COMPONENT" pour montrer où peuvent apparaître les extensions de propriété et de composant enregistrées et expérimentales. Les tableaux n'affichent pas les restrictions de paramètres de propriété. Ces restrictions sont définies dans la [RFC5545].

3.1 Tableaux des restrictions communes aux composants

3.1.1 VCALENDAR

Le tableau de restrictions ci-dessous s'applique aux propriétés de l'objet iCalendar. C'est-à-dire, les propriétés à la portée la plus externe.

Contraintes sur les propriétés dans un composant VCALENDAR

Composant/propriété	Présence	Commentaire
CALSCALE	0 ou 1	
PRODID	1	
VERSION	1	La valeur DOIT être 2.0.
IANA-PROPERTY	0+	
X-PROPERTY	0+	

3.1.2 VTIMEZONE

D'autres composants peuvent se référer aux composants "VTIMEZONE" via un paramètre "TZID" sur un type de valeur "DATETIME". Les restrictions de propriété dans le tableau ci-dessous s'appliquent à tout composant "VTIMEZONE" dans un message iTIP.

Contraintes pour les composants VTIMEZONE

Composant/propriété	Présence	Commentaire
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
DAYLIGHT	0+	DOIT être un ou plusieurs de STANDARD ou DAYLIGHT.
COMMENT	0+	
DTSTART	1	DOIT être en format d'heure locale.
RDATE	0+	
RRULE	0 ou 1	
TZNAME	0+	
TZOFFSETFROM	1	
TZOFFSETTO	1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
LAST-MODIFIED	0 ou 1	
STANDARD	0+	DOIT être un ou plusieurs de STANDARD ou DAYLIGHT.
COMMENT	0+	
DTSTART	1	DOIT être en format d'heure locale.
RDATE	0+	Si présent, RRULE NE DOIT PAS être présent.
RRULE	0 ou 1	Si présent, RDATE NE DOIT PAS être présent.
TZNAME	0+	
TZOFFSETFROM	1	
TZOFFSETTO	1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
TZID	1	
TZURL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	

3.1.3 VALARM

Les restrictions de propriété dans le tableau ci-dessous s'appliquent à tout composant "VALARM" dans un message iTIP.

Contraintes pour les composants VALARM

Composant/propriété	Présence	Commentaire
VALARM	0+	
ACTION	1	
ATTACH	0+	
ATTENDEE	0+	
DESCRIPTION	0 ou 1	
DURATION	0 ou 1	Si présent, REPEAT DOIT être présent.
REPEAT	0 ou 1	Si présent, DURATION DOIT être présent.
SUMMARY	0 ou 1	
TRIGGER	1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	

3.2 Méthodes pour les composants de calendrier VEVENT

Ce paragraphe définit les restrictions d'ensemble de propriétés pour les types de méthodes applicables au composant de calendrier "VEVENT". Chaque méthode est définie en utilisant un tableau qui précise les contraintes de propriété que définit la méthode particulière.

Voici la récapitulation des méthodes définies pour le composant de calendrier "VEVENT".

Méthode	Description
PUBLISH	Post notification d'un événement. Utilisé principalement comme méthode d'annonce de l'existence d'un événement.
REQUEST	Fait une demande d'un événement. C'est une invitation explicite à un ou plusieurs participants. Les demandes d'événement sont aussi utilisées pour mettre à jour ou changer un événement existant. Les clients qui ne peuvent pas traiter REQUEST PEUVENT dégrader l'événement pour le voir comme un PUBLISH.
REPLY	Répond à une demande d'événement. Les clients peuvent régler leur état (PARTSTAT) à ACCEPTED, DECLINED, TENTATIVE, ou DELEGATED.
ADD	Ajoute une ou plusieurs instances à un événement existant.
CANCEL	Annule une ou plusieurs instances d'un événement existant.
REFRESH	Un participant envoie à un organisateur une demande de la dernière version d'un événement à envoyer à nouveau au demandeur.
COUNTER	Contre une REQUEST avec un autre proposition. Envoyé par un participant à l'organisateur.
DECLINECOUNTER	Refuse une contre proposition. Envoyé à un participant par l'organisateur.

3.2.1 PUBLISH

La méthode "PUBLISH" dans un composant de calendrier "VEVENT" est un envoi non sollicité d'un objet iCalendar. Tout CU peut ajouter des composants publiés à son calendrier. L'"Organisateur" DOIT être présent dans un composant iCalendar publié. Les "Participants" NE DOIVENT PAS être présents. Son usage prévu est d'encapsuler un événement arbitraire comme objet iCalendar. L'"Organisateur" peut ensuite mettre à jour (avec une autre méthode "PUBLISH") ajouter des instances (avec une méthode "ADD") ou annuler (avec une méthode "CANCEL") un composant de calendrier "VEVENT" publié précédemment.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:PUBLISH d'un VEVENT

Composant/propriété	Présence	Commentaire
METHOD	1	DOIT être égal à PUBLISH.
VEVENT	1+	
DTSTAMP	1	
DTSTART	1	
ORGANIZER	1	
SUMMARY	1	Peut être nul.
UID	1	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
SEQUENCE	ou 1	DOIT être présent si valeur est supérieure à 0 ; PEUT être présent si 0.
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
CONTACT	0 ou 1	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	Peut être nul.
DTEND	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DTEND NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PRIORITY	0 ou 1	

RDATE	0+	
RELATED-TO	0+	
RESOURCES	0+	
RRULE	0 ou 1	
STATUS	0 ou 1	PEUT être un de TENTATIVE/CONFIRMED/CANCELLED.
TRANSP	0 ou 1	
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
ATTENDEE	0	
REQUEST-STATUS	0	
VALARM	0+	
VFREEBUSY	0	
VJOURNAL	0	
VTODO	0	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	

3.2.2 REQUEST

La méthode "REQUEST" dans un composant "VEVENT" fournit les fonctions de programmation suivantes :

- o Invite les "Participants" à un événement.
- o Reprogramme un événement existant.
- o Réponse à une demande "REFRESH".
- o Mise à jour des détails d'un événement existant, sans le reprogrammer.
- o Mise à jour de l'état des "Participants" à un événement existant, sans le reprogrammer.
- o Reconfirmer un événement existant, sans le reprogrammer.
- o Transmettre un "VEVENT" à un autre CU non invité.
- o Pour un composant de calendrier "VEVENT" existant, délègue le rôle de "Participant" à un autre CU.
- o Pour un composant de calendrier "VEVENT" existant, change le rôle de "Organisateur" sur un autre CU.

L'"Organisateur" génère le "REQUEST". Les receveurs de la méthode "REQUEST" sont les CU invités à l'événement, les "Participants". Les "Participants" utilisent la méthode "REPLY" pour porter l'état de participant à l'"Organisateur".

Les propriétés "UID" et "SEQUENCE" sont utilisées pour distinguer les diverses utilisations de la méthode "REQUEST". Si la valeur de propriété "UID" dans la "REQUEST" n'est pas trouvée sur le calendrier du receveur, alors la "REQUEST" est pour un nouveau composant de calendrier "VEVENT". Si la valeur de propriété "UID" est trouvée sur le calendrier du receveur, alors la "REQUEST" est pour une reprogrammation, une mise à jour, ou une reconfirmation du composant de calendrier "VEVENT".

Pour la méthode "REQUEST", plusieurs composants "VEVENT" dans un seul objet iCalendar ne sont permis que pour les composants qui ont la même propriété "UID". C'est-à-dire, une série d'événements récurrents peut avoir des informations spécifiques de l'instance. Dans ce cas, plusieurs composants "VEVENT" sont nécessaires pour exprimer la série entière.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:REQUEST d'un VEVENT

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être REQUEST.
VEVENT	1+	Tous les composants DOIVENT avoir le même UID.
ATTENDEE	1+	
DTSTAMP	1	
DTSTART	1	
ORGANIZER	1	
SEQUENCE	0 ou 1	DOIT être présent si valeur est supérieure à 0; PEUT être présent si 0.
SUMMARY	1	Peut être nul.
UID	1	
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	

COMMENT	0+	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	Peut être nul.
DTEND	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DTEND NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PRIORITY	0 ou 1	
RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
REQUEST-STATUS	0	
RESOURCES	0+	
RRULE	0 ou 1	
STATUS	0 ou 1	PEUT être un de TENTATIVE/CONFIRMED.
TRANSP	0 ou 1	
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
VALARM	0+	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
ANA-COMPONENT	0+	
X-COMPONENT	0+	
VFREEBUSY	0	
VJOURNAL	0	
VTODO	0	

3.2.2.1 Reprogrammation d'un événement

La méthode "REQUEST" peut être utilisée pour reprogrammer un événement. Un événement reprogrammé implique un changement à l'événement existant en termes de son heure ou ses intervalles de récurrence et possiblement sa localisation ou description. Si le CUA receveur d'une méthode "REQUEST" trouve que la valeur de propriété "UID" existe déjà sur le calendrier mais que la valeur de propriété "SEQUENCE" (ou "DTSTAMP") dans la méthode "REQUEST" est supérieure à la valeur pour l'événement existant, alors la méthode "REQUEST" décrit une reprogrammation de l'événement.

3.2.2.2 Mise à jour ou reconfirmation d'un événement

La méthode "REQUEST" peut être utilisée pour mettre à jour ou reconfirmer un événement. Une mise à jour d'un événement existant n'implique pas de changement à l'heure ou aux intervalles de récurrence, et pourrait ne pas impliquer de changement de la localisation ou de la description pour l'événement. Si le CUA receveur d'une méthode "REQUEST" trouve que la valeur de propriété "UID" existe déjà sur le calendrier et que la valeur de propriété "SEQUENCE" dans la "REQUEST" est la même que la valeur pour l'événement existant, alors la méthode "REQUEST" décrit une mise à jour des détails de l'événement, mais pas une reprogrammation de l'événement.

La méthode de mise à jour "REQUEST" est la réponse appropriée à une méthode "REFRESH" envoyée d'un "Participant" à l'"Organisateur" d'un événement.

L'"Organisateur" d'un événement peut aussi envoyer des méthodes "REQUEST" non sollicitées. Les méthodes "REQUEST" non sollicitées peuvent être utilisées pour mettre à jour les détails de l'événement sans le reprogrammer, pour mettre à jour le paramètre "PARTSTAT" des "Participants", ou pour reconfirmer l'événement.

3.2.2.3 Délégation d'un événement à un autre CU

Certains systèmes de calendrier et de programmation permettent aux "Participants" de déléguer leur présence à un événement à un autre "Utilisateur de calendrier". iTIP prend en charge ce concept en utilisant le flux d'actions suivant. Tous "Participant" peut déléguer son droit de participer à un calendrier "VEVENT" à un autre CU. L'implication est que le délégué participe à la place du "participant" original, et NON en plus du "Participant". Le déléguant DOIT notifier cette

action à l'"Organisateur" en utilisant les étapes mentionnées ci-dessous. Les mises en œuvre peut prendre en charge ou interdire la délégation comme il leur convient. Par exemple, certaines mises en œuvre peuvent restreindre un délégué à déléguer une "REQUEST" à un autre CU.

Le "Déléguant" d'un événement transmet le "REQUEST" existant au "Délégué". La méthode "REQUEST" DOIT inclure une propriété "ATTENDEE" avec l'adresse de calendrier du "Délégué". Le "Déléguant" DOIT aussi envoyer une méthode "REPLY" à l'"Organisateur" avec la valeur du paramètre "PARTSTAT" de la propriété "ATTENDEE" du "Déléguant" réglée à "DELEGATED". De plus, le paramètre "DELEGATED-TO" DOIT être inclus avec l'adresse de calendrier du "Délégué". Aussi, une nouvelle propriété "ATTENDEE" pour le "Délégué" DOIT être incluse et doit spécifier l'adresse d'utilisateur de calendrier réglée dans le paramètre "DELEGATED-TO", comme ci-dessus.

En réponse à la demande, le "Délégué" DOIT envoyer une méthode "REPLY" à l'"Organisateur", et facultativement au "Déléguant". La méthode "REPLY" DEVRAIT inclure la propriété "ATTENDEE" avec la valeur de paramètre "DELEGATED-FROM" de l'adresse de calendrier du "Déléguant".

Le "Déléguant" peut continuer de recevoir les mises à jour de l'événement même si il ne va pas y participer. Ceci est accompli par le réglage par le "Déléguant" de son attribut "role" à "NON-PARTICIPANT" dans le "REPLY" à l'"Organisateur".

3.2.2.4 Changement d'organisateur

Il peut se produire une situation où l'"Organisateur" d'un "VEVENT" n'est plus capable de tenir le rôle "d'Organisateur" et abdique sans passer le rôle "d'Organisateur" à quelqu'un d'autre. Quand cela se produit, les "Participants" au "VEVENT" peuvent utiliser des mécanismes hors bande pour se communiquer la situation et se mettre d'accord sur un nouvel "Organisateur". Le nouvel "Organisateur" devrait alors envoyer une nouvelle "REQUEST" avec une version modifiée du "VEVENT" dans laquelle le numéro de "SEQUENCE" a été incrémenté et la propriété "ORGANIZER" a été changée pour le nouvel "Organisateur".

3.2.2.5 Envoi au nom de l'organisateur

Un certain nombre de scénarios prennent en charge le besoin qu'un "Utilisateur de calendrier" agisse au nom de l'"Organisateur" sans changement explicite de rôle. Ce pourrait être le cas si le CU désigné comme "Organisateur" est malade ou incapable d'effectuer les tâches associées à cette fonction. Dans ce cas, iTIP prend en charge la notion d'un CU agissant au nom d'un autre. En utilisant le paramètre "SENT-BY", un "Utilisateur de calendrier" pourrait envoyer une "REQUEST" de "VEVENT" mis à jour. Dans le cas où un CU envoie au nom d'un autre CU, les réponses de "Participant" sont quand même redirigées vers le CU désigné comme "Organisateur".

3.2.2.6 Transmission à un CU non invité

Un "participant" invité à un composant de calendrier "VEVENT" peut envoyer le composant de calendrier "VEVENT" à un autre nouveau CU non associé précédemment au composant de calendrier "VEVENT". Le "Participant" actuel invité au composant de calendrier "VEVENT" fait cela en transmettant la méthode "REQUEST" originale au nouveau CU. Le nouveau CU peut envoyer une "REPLY" à l'"Organisateur" du composant de calendrier "VEVENT". La réponse contient une propriété "ATTENDEE" pour le nouveau CU.

L'"Organisateur" décide en fin de compte si le nouveau CU participe ou non à l'événement et n'est obligé à rien par une "REPLY" provenant d'un nouveau CU (non invité). Si l'"Organisateur" ne veut pas que le nouveau CU fasse partie de l'événement, la nouvelle propriété "ATTENDEE" n'est pas ajoutée au composant de calendrier "VEVENT". L'"Organisateur" PEUT envoyer au CU un message "CANCEL" pour indiquer qu'il ne va pas être ajouté à l'événement. Si l'"Organisateur" décide d'ajouter le nouveau CU, la nouvelle propriété "ATTENDEE" est ajoutée au composant de calendrier "VEVENT". De plus, l'"Organisateur" est libre de changer tout paramètre de propriété "ATTENDEE" des valeurs fournies par le nouveau CU en quelque chose que l'"Organisateur" considère approprié. L'"Organisateur" DEVRAIT envoyer au nouveau CU un message "REQUEST" pour l'informer qu'il a été ajouté.

Quand il transmet un "REQUEST" à un autre CU, le "Participant" qui transmet NE DOIT PAS faire de changement au message original.

3.2.2.7 Mise à jour de l'état de Participant

L'"Organisateur" d'un événement peut aussi demander la mise à jour de l'état d'un ou plusieurs "Participants". L'"Organisateur" envoie une méthode "REQUEST" au "Participant" et règle le paramètre de propriété

"ATTENDEE;RSVP=TRUE". La propriété "SEQUENCE" pour l'événement n'est pas changée. Un receveur va déterminer que le seul changement dans la "REQUEST" est que son paramètre de propriété "RSVP" indique une demande de mise à jour d'état. Le receveur DEVRAIT répondre avec une méthode "REPLY" indiquant son état actuel par rapport à la "REQUEST".

3.2.3 REPLY

La méthode "REPLY" dans un composant de calendrier "VEVENT" est utilisée pour répondre (par exemple, accepter ou refuser) à une "REQUEST" ou pour répondre à une demande de délégation. Quand elle est utilisée pour fournir une réponse de délégation, le "Déléguant" DEVRAIT inclure l'adresse de calendrier du "Délégué" sur le paramètre de propriété "DELEGATED-TO" de la propriété "ATTENDEE" du "Déléguant". Le "Délégué" DEVRAIT inclure l'adresse de calendrier du "Déléguant" sur le paramètre de propriété "DELEGATED-FROM" de la propriété "ATTENDEE" du "Délégué".

La méthode "REPLY" est aussi utilisée quand le traitement d'une "REQUEST" échoue. Selon la valeur de la propriété "REQUEST-STATUS", aucune action de programmation ne peut avoir été effectuée.

L'"Organisateur" d'un événement peut recevoir la méthode "REPLY" d'un CU en dehors de la "REQUEST" originale. Par exemple, une "REPLY" peut être reçue d'un "Délégué" à un événement. De plus, la méthode "REPLY" peut être reçue d'un CU inconnu (un "Party Crasher" (*trouble-fête*)). Ce "Participant" non invité peut être accepté, ou l'"Organisateur" peut annuler l'événement pour le "Participant" non invité en envoyant une méthode "CANCEL" au "Participant" non invité.

Un "Participant" PEUT inclure un message à l'"Organisateur" en utilisant la propriété "COMMENT". Par exemple, si l'utilisateur indique une tentative d'acceptation et veut en faire savoir la raison à l'"Organisateur", celle-ci peut être exprimée dans la valeur de propriété "COMMENT".

L'"Organisateur" peut aussi recevoir une "REPLY" d'un CU au nom d'un autre. Comme dans le scénario mentionné ci-dessus pour l'"Organisateur", les "Participants" peuvent avoir un autre CU qui répond en leur nom. Ceci est fait en utilisant le paramètre "SENT-BY".

Les propriétés facultatives mentionnées dans le tableau ci-dessous (celles marquées "0+" ou "0 ou 1") NE DOIVENT PAS être changées par rapport à la demande d'origine. Si des changements de propriété sont désirés, le message "COUNTER" doit être utilisé.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:REPLY d'un VEVENT

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être REPLY.
VEVENT	1+	Tous les composants DOIVENT avoir le même UID.
ATTENDEE	1	DOIT être l'adresse du participant qui répond.
DTSTAMP	1	
ORGANIZER	1	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
UID	1	DOIT être l'UID de la REQUEST originale.
SEQUENCE	0 ou 1	Si non zéro, DOIT être le numéro de séquence de la REQUEST originale. PEUT être présent si 0.
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	
DTEND	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DTSTART	0 ou 1	
DURATION	0 ou 1	Si présent, DTEND NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	

LOCATION	0 ou 1	
PRIORITY	0 ou 1	
RDATE	0+	
RELATED-TO	0+	
RESOURCES	0+	
REQUEST-STATUS	0+	
RRULE	0 ou 1	
STATUS	0 ou 1	
SUMMARY	0 ou 1	
TRANSP	0 ou 1	
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
VALARM	0	
VTIMEZONE	0 ou 1	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VFREEBUSY	0	
VJOURNAL	0	
VTODO	0	

3.2.4 ADD

La méthode "ADD" permet à l'"Organisateur" d'ajouter une ou plusieurs nouvelles instances à un "VEVENT" existant en utilisant un seul message iTIP sans avoir à envoyer le "VEVENT" entier avec toutes les données d'instance existantes, comme il devrait le faire si la méthode "REQUEST" était utilisée.

Le "UID" doit être celui de l'événement existant. Si la valeur de propriété "UID" dans le "ADD" n'est pas trouvée dans le calendrier du receveur, celui-ci DEVRAIT envoyer un "REFRESH" à l'"Organisateur" afin d'être mis à jour avec la dernière version du "VEVENT". Si une mise en œuvre de "participant" ne prend pas en charge la méthode "ADD", elle devrait répondre avec une valeur "REQUEST-STATUS" de 3.14 et demander un "REFRESH".

Quand il traite un message "ADD", le "participant" traite chaque composant dans le message "ADD" comme si il était référencé via un "RDATE" dans le composant principal.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:ADD d'un VEVENT

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être ADD.
VEVENT	1	
DTSTAMP	1	
DTSTART	1	
ORGANIZER	1	
SEQUENCE	1	DOIT être supérieure à 0.
SUMMARY	1	PEUT être nul.
UID	1	DOIT correspondre à celle de l'événement original.
ATTACH	0+	
ATTENDEE	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	Peut être nul.
DTEND	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DTEND NE DOIT PAS être présent.
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PRIORITY	0 ou 1	

RELATED-TO	0+	
RESOURCES	0+	
STATUS	0 ou 1	PEUT être un de TENTATIVE/CONFIRMED.
TRANSP	0 ou 1	
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
EXDATE	0	
RECURRENCE-ID	0	
REQUEST-STATUS	0	
RDATE	0	
RRULE	0	
VALARM	0+	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VFREEBUSY	0	
VTODO	0	
VJOURNAL	0	

3.2.5 CANCEL

La méthode "CANCEL" dans un composant de calendrier "VEVENT" est utilisée pour envoyer un avis d'annulation d'une demande d'événement existant aux "Participants" affectés. Le message est envoyé par l'"Organisateur" de l'événement. Pour un événement récurrent, l'événement entier ou des instances d'un événement peuvent être annulées. Pour annuler la gamme complète d'un événement récurrent, la valeur de propriété "UID" pour l'événement DOIT être spécifiée et un "RECURRENCE-ID" NE DOIT PAS être spécifié dans la méthode "CANCEL". Afin d'annuler une instance individuelle de l'événement, la valeur de propriété "RECURRENCE-ID" pour l'événement DOIT être spécifiée dans la méthode "CANCEL".

Il y a deux options pour annuler une séquence d'instances d'un composant de calendrier "VEVENT" récurrent :

- La propriété "RECURRENCE-ID" pour une instance dans la séquence DOIT être spécifiée avec la valeur de paramètre de propriété "RANGE" de "THISANDFUTURE" pour indiquer l'annulation du composant de calendrier "VEVENT" spécifié et de toutes les instances après lui.
- Des instances de récurrence individuelles peuvent être annulées en spécifiant plusieurs composants "VEVENT" ayant chacun une propriété "RECURRENCE-ID" correspondant à une des instances à annuler.

L'"Organisateur" DOIT envoyer un message "CANCEL" à chaque "Participant" affecté par l'annulation. Ceci peut être fait en utilisant un seul message "CANCEL" pour tous les "Participants" ou en utilisant plusieurs messages avec différents sous ensembles de "Participants" affectés dans chacun.

Quand un "VEVENT" est annulé, la valeur de propriété "SEQUENCE" DOIT être incrémentée comme décrit au paragraphe 2.1.4.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:CANCEL d'un VEVENT

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être CANCEL.
VEVENT	1+	Tous doivent avoir le même UID.
ATTENDEE	0+	DOIT inclure certains ou tous les participants retirés de l'événement. DOIT inclure certains ou tous les participants si l'événement entier est annulé.
DTSTAMP	1	
ORGANIZER	1	
SEQUENCE	1	
UID	1	DOIT être l'UID de la REQUEST originale.
COMMENT	0+	
ATTACH	0+	
CATEGORIES	0+	

CLASS	0 ou 1	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	
DTEND	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DTSTART	0 ou 1	
DURATION	0 ou 1	Si présent, DTEND NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PRIORITY	0 ou 1	
RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
RESOURCES	0+	
RRULE	0 ou 1	
STATUS	0 ou 1	DOIT être réglé à CANCELLED pour annuler l'événement entier. Si il n'invite pas des participants spécifiques, il NE DOIT alors PAS être inclus.
SUMMARY	0 ou 1	
TRANSP	0 ou 1	
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
REQUEST-STATUS	0	
VALARM	0	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VTODO	0	
VJOURNAL	0	
VFREEBUSY	0	

3.2.6 REFRESH

La méthode "REFRESH" dans un composant de calendrier "VEVENT" est utilisée par les "Participants" à un événement existant pour demander une description mise à jour à l'"Organisateur" de l'événement. La méthode "REFRESH" doit spécifier la propriété "UID" de l'événement à mettre à jour. Une instance récurrente d'un événement peut être demandée en spécifiant la propriété "RECURRENCE-ID" correspondant à l'événement associé. L'"Organisateur" répond avec la dernière description et version de l'événement.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:REFRESH d'un VEVENT

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être REFRESH.
VEVENT	1	
ATTENDEE	1	DOIT être l'adresse du demandeur.
DTSTAMP	1	
ORGANIZER	1	
UID	1	DOIT être l'UID associé à la REQUEST originale.
COMMENT	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
IANA-PROPERTY	0+	
X-PROPERTY	0+	
ATTACH	0	
CATEGORIES	0	
CLASS	0	
CONTACT	0	

CREATED	0
DESCRIPTION	0
DTEND	0
DTSTART	0
DURATION	0
EXDATE	0
GEO	0
LAST-MODIFIED	0
LOCATION	0
PRIORITY	0
RDATE	0
RELATED-TO	0
REQUEST-STATUS	0
RESOURCES	0
RRULE	0
SEQUENCE	0
STATUS	0
SUMMARY	0
TRANSP	0
URL	0
VALARM	0
VTIMEZONE	0+
IANA-COMPONENT	0+
X-COMPONENT	0+
VTODO	0
VJOURNAL	0
VFREEBUSY	0

3.2.7 COUNTER

La méthode "COUNTER" pour un composant de calendrier "VEVENT" est utilisée par un "Participant" à un événement existant pour soumettre à l'"Organisateur" une contre proposition pour l'événement. Le "Participant" envoie ce message à l'"Organisateur" de l'événement.

La contre proposition est un objet iCalendar consistant en un composant de calendrier "VEVENT" qui fournit une description complète de l'événement de remplacement.

L'"Organisateur" rejette la contre proposition en envoyant au "Participant" une méthode "DECLINECOUNTER". L'"Organisateur" accepte la contre proposition en reprogrammant l'événement comme décrit au paragraphe 3.2.2.1, "Reprogrammation d'un événement". Le CUA de l'"Organisateur" DEVRAIT envoyer un message "REQUEST" à tous les "Participants" affectés par tout changement déclenché par une "COUNTER" acceptée.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:COUNTER d'un VEVENT

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être COUNTER.
VEVENT	1	
DTSTAMP	1	
DTSTART	1	
ORGANIZER	1	DOIT être l'organisateur de l'événement original.
SEQUENCE	1	DOIT faire écho du numéro de SEQUENCE original. DOIT être présent si non zéro. PEUT être présent si zéro.
SUMMARY	1	Peut être nul.
UID	1	DOIT être l'UID associé à la REQUEST contrée.
ATTACH	0+	
ATTENDEE	0+	Peut aussi être utilisé pour proposer d'autres participants.
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
CONTACT	0+	

CREATED	0 ou 1	
DESCRIPTION	0 ou 1	
DTEND	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DTEND NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PRIORITY	0 ou 1	
RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
REQUEST-STATUS	0+	
RESOURCES	0+	
RRULE	0 ou 1	
STATUS	0 ou 1	La valeur doit être une de CONFIRMED/TENTATIVE/CANCELLED.
TRANSP	0 ou 1	
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
VALARM	0+	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VTODO	0	
VJOURNAL	0	
VFREEBUSY	0	

3.2.8 DECLINECOUNTER

La méthode "DECLINECOUNTER" dans un composant de calendrier "VEVENT" est utilisée par l'"Organisateur" d'un événement pour rejeter une contre proposition soumise par un "Participant". L'"Organisateur" doit envoyer le message "DECLINECOUNTER" au "Participant" qui a envoyé la méthode "COUNTER" à l'"Organisateur".

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:DECLINECOUNTER d'un VEVENT

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être DECLINECOUNTER.
VEVENT	1+	Tous les composants DOIVENT avoir le même UID.
ATTENDEE	1+	DOIT être pour tous les participants.
DTSTAMP	1	
ORGANIZER	1	
SEQUENCE	1	DOIT faire écho au numéro de SEQUENCE original.
UID	1	DOIT faire écho à l'UID original.
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	Peut être nul.
DTSTART	0 ou 1	
DTEND	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DTEND NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PRIORITY	0 ou 1	

RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
REQUEST-STATUS	0+	
RESOURCES	0+	
RRULE	0 ou 1	
STATUS	0 ou 1	PEUT être un de TENTATIVE/CONFIRMED.
SUMMARY	0 ou 1	Peut être nul.
TRANSP	0 ou 1	
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VALARM	0	
VFREEBUSY	0	
VJOURNAL	0	
VTODO	0	

3.3 Méthodes pour les composants VFREEBUSY

Ce paragraphe définit l'ensemble de propriétés pour les méthodes qui sont applicables au composant de calendrier "VFREEBUSY". Chacune des méthodes est définie en utilisant un tableau de restrictions.

Le présent document traite seulement du transfert des informations des heures occupées. Les applications qui désirent les informations de temps libres DOIVENT déduire cela des informations d'heures occupées disponibles.

La valeur de propriété "FREEBUSY" PEUT inclure une liste de valeurs, séparées par le caractère virgule (décimal 44 en US-ASCII). Autrement, plusieurs périodes de temps d'occupation PEUVENT être spécifiées avec plusieurs instances de la propriété "FREEBUSY". Les deux formes DOIVENT être prises en charge par les mises en œuvre qui se conforment au présent document. Les périodes de temps occupées dupliquées NE DEVRAIENT PAS être spécifiées dans un objet iCalendar. Cependant, deux périodes de temps d'occupation différentes PEUVENT se chevaucher.

Les propriétés "FREEBUSY" DEVRAIENT être triées de telle façon que leurs valeurs soient en ordre ascendant, sur la base de l'heure de début et ensuite de l'heure de fin, avec les périodes les plus tôt en premier. Par exemple, les informations de temps d'occupation d'aujourd'hui devraient apparaître après les informations de temps d'occupation d'hier. Et les informations de temps d'occupation pour la présente demie heure devraient apparaître après les heures d'occupation de plus tôt dans la journée. Les périodes de temps d'occupation peuvent aussi dépasser les limites d'une journée.

On résume ci-dessous les méthodes qui sont définies pour le composant de calendrier "VFREEBUSY".

Méthode Description

PUBLISH	Publie les données de temps d'occupation non sollicitées.
REQUEST	Demandes les données de temps d'occupation.
REPLY	Répond à une demande des temps d'occupation.

3.3.1 PUBLISH

La méthode "PUBLISH" dans un composant de calendrier "VFREEBUSY" est utilisée pour publier les données de temps d'occupation. La méthode peut être envoyé d'un CU à tout autre. L'objet de la méthode est de donner le moyen d'envoyer des données de temps d'occupation non sollicitées. C'est-à-dire, les données de temps d'occupation ne sont pas envoyées comme une "REPLY" à la réception d'une méthode "REQUEST".

La propriété "ORGANIZER" DOIT être spécifiée dans les informations de temps d'occupation. La valeur est l'adresse du CU de l'origine des informations de temps d'occupation.

Les informations de temps d'occupation au sein de l'objet iCalendar PEUVENT être groupées en plus d'un composant de calendrier "VFREEBUSY". Cette capacité permet que les périodes de temps d'occupation soient groupées en accord avec

une périodicité commune, comme une semaine, un mois, ou une année de calendrier. Dans ce cas, chaque composant de calendrier "VFREEBUSY" DOIT inclure les propriétés "ORGANIZER", "DTSTART", et "DTEND" afin de spécifier la source des informations de temps d'occupation et l'intervalle de date et heure que couvrent les informations de temps d'occupation.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:PUBLISH d'un VFREEBUSY

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être PUBLISH.
VFREEBUSY	1+	
DTSTAMP	1	
DTSTART	1	Les valeurs de date et heure doivent être en UTC.
DTEND	1	Les valeurs de date et heure doivent être en UTC.
FREEBUSY	0+	DOIT être BUSYTIME. Plusieurs instances sont permises. Plusieurs instances DEVRAIENT être rangées en ordre ascendant.
ORGANIZER	1	DOIT contenir l'adresse de l'origine des données de temps d'occupation.
UID	1	
COMMENT	0+	
CONTACT	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
URL	0 ou 1	Spécifie l'URL des temps d'occupation.
ATTENDEE	0	
DURATION	0	
REQUEST-STATUS	0	
VALARM	0	
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VTODO	0	
VJOURNAL	0	
VTIMEZONE	0	

3.3.2 REQUEST

La méthode "REQUEST" dans un composant de calendrier "VFREEBUSY" est utilisée pour demander à un "Utilisateur de calendrier" ses informations de temps d'occupation. La demande peut être pour des informations de temps d'occupation bordées par un intervalle spécifique de date et d'heure.

Ce message permet seulement des demandes d'informations de temps d'occupation. Le message est envoyé d'un "Utilisateur de calendrier" qui demande les informations de temps d'occupation d'un ou plusieurs receveurs.

Si le générateur de la méthode "REQUEST" n'est pas autorisé à faire de demande de temps d'occupation sur le système de calendrier du receveur, un message d'exception DEVRAIT alors être retourné dans une méthode "REPLY", mais aucune donnée de temps d'occupation n'a besoin d'être retournée.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:REQUEST d'un VFREEBUSY

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être REQUEST.
VFREEBUSY	1	
ATTENDEE	1+	Contient les adresses d'utilisateur de calendrier des "Utilisateurs de calendrier" dont les temps d'occupation sont demandés.
DTEND	1	Les valeurs de date et heure doivent être en UTC.
DTSTAMP	1	
DTSTART	1	Les valeurs de date et heure doivent être en UTC.
ORGANIZER	1	DOIT être l'adresse de l'origine de la demande.
UID	1	
COMMENT	0+	

CONTACT	0 ou
IANA-PROPERTY	0+
X-PROPERTY	0+
FREEBUSY	0
DURATION	0
REQUEST-STATUS	0
URL	0
VALARM	0
IANA-COMPONENT	0+
X-COMPONENT	0+
VEVENT	0
VTODO	0
VJOURNAL	0
VTIMEZONE	0

3.3.3 REPLY

La méthode "REPLY" dans un composant de calendrier "VFREEBUSY" est utilisée pour répondre à une demande de temps d'occupation. La méthode est envoyée par le receveur d'une demande de temps d'occupation au générateur de la demande.

La méthode "REPLY" peut aussi être utilisée pour répondre à une méthode "REQUEST" qui n'a pas réussi. Selon la valeur de la "REQUEST-STATUS", aucune information de temps d'occupation peut n'être retournée.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:REPLY d'un VFREEBUSY

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être REPLY.
VFREEBUSY	1	
ATTENDEE	1	DOIT être l'adresse du participant qui répond.
DTSTAMP	1	
DTEND	1	Les valeurs de date et heure doivent être en UTC.
DTSTART	1	Les valeurs de date et heure doivent être en UTC.
FREEBUSY	0+	DOIT être BUSYTIME. Plusieurs instances sont permises. Plusieurs instances DEVRAIENT être rangées en ordre ascendant.
ORGANIZER	1	DOIT être l'adresse du générateur de la demande.
UID	1	DOIT être l'UID de la REQUEST originale.
COMMENT	0+	
CONTACT	0 ou 1	
REQUEST-STATUS	0+	
URL	0 ou 1	Spécifie l'URL des temps d'occupation.
IANA-PROPERTY	0+	
X-PROPERTY	0+	
DURATION	0	
SEQUENCE	0	
VALARM	0	
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VTODO	0	
VJOURNAL	0	
VTIMEZONE	0	

3.4 Méthodes pour les composants VTODO

Ce paragraphe définit l'ensemble de propriétés pour les méthodes qui sont applicables au composant de calendrier "VTODO". Chacune des méthodes est définie en utilisant un tableau de restrictions qui spécifie les contraintes de propriété qui définissent la méthode particulière.

On résume ci-dessous les méthodes qui sont définies pour le composant de calendrier "VTODO".

Méthode	Description
PUBLISH	Post notification d'un VTODO. Utilisé principalement comme méthode d'annonce de l'existence d'un VTODO.
REQUEST	Alloue un VTODO. C'est une allocation explicite à un ou plusieurs utilisateurs de calendrier. La méthode REQUEST est aussi utilisée pour mettre à jour ou changer un VTODO existant. Les clients qui ne peuvent pas traiter la REQUEST PEUVENT dégrader la méthode pour la traiter comme PUBLISH.
REPLY	Répond à une demande VTODO. Les participants PEUVENT régler PARTSTAT à ACCEPTED, DECLINED, TENTATIVE, DELEGATED, PARTIAL, et COMPLETED.
ADD	Ajoute une ou plusieurs instances à une chose à faire existante.
CANCEL	Annule une ou plusieurs instances d'une chose à faire existante.
REFRESH	Demande envoyée à un organisateur de VTODO pour lui demander la dernière version d'un VTODO.
COUNTER	Contre une REQUEST avec une proposition de remplacement.
DECLINECOUNTER	Refuse une contre proposition d'un participant.

3.4.1 PUBLISH

La méthode "PUBLISH" dans un composant de calendrier "VTODO" n'a pas de réponse associée. C'est simplement un envoi d'un objet iCalendar qui peut être ajouté à un calendrier. Elle DOIT avoir un "Organisateur". Elle NE DOIT PAS avoir de "Participant". Son utilisation prévue est l'encapsulation d'un composant de calendrier "VTODO" arbitraire comme objet iCalendar. L'"Organisateur" PEUT ensuite mettre à jour (avec une autre méthode "PUBLISH") ajouter des instances (avec une méthode "ADD") ou annuler (avec une méthode "CANCEL") un composant de calendrier "VTODO" précédemment publié.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:PUBLISH d'un VTODO

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être PUBLISH.
VTODO	1+	
DTSTAMP	1	
DTSTART	1	
ORGANIZER	1	
PRIORITY	1	
SEQUENCE	0 ou 1	DOIT être présent si la valeur est supérieure à 0 ; PEUT être présent si 0.
SUMMARY	1	Peut être nul.
UID	1	
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
COMPLETED	0 ou 1	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	Peut être nul.
DUE	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DUE NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PERCENT-COMPLETE	0 ou 1	
RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
RESOURCES	0+	
RRULE	0 ou 1	

STATUS	0 ou 1	PEUT être un de COMPLETED/NEEDS-ACTION/IN-PROCESS/CANCELLED.
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
ATTENDEE	0	
REQUEST-STATUS	0	
VALARM	0+	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VFREEBUSY	0	
VEVENT	0	
VJOURNAL	0	

3.4.2 REQUEST

La méthode "REQUEST" dans un composant de calendrier "VTODO" fournit les fonctions de programmation suivantes :

- o Allouer une chose à faire à un ou plusieurs "Utilisateurs de calendrier".
- o Reprogrammer une chose à faire existante.
- o Mettre à jour les détails d'une chose à faire existante, sans la reprogrammer.
- o Mettre à jour l'état d'achèvement des "Participants" à une chose à faire existante, sans la reprogrammer.
- o Reconfirmer une chose à faire existante, sans la reprogrammer.
- o Déléguer/réallouer une chose à faire existante à un autre "Utilisateur de calendrier".

Les "Utilisateurs de calendrier" alloués sont identifiés dans le composant de calendrier "VTODO" par les séquences individuelles de valeur de propriété "ATTENDEE;ROLE=REQ-PARTICIPANT".

Normalement, le générateur d'une "REQUEST" est l'"Organisateur" de la chose à faire, et le receveur d'une "REQUEST" est l'"Utilisateur de calendrier" auquel la chose à faire est allouée. Le "Participant" utilise la méthode "REPLY" pour porter son acceptation et l'état d'achèvement à l'"Organisateur" de la "REQUEST".

Les propriétés "UID", "SEQUENCE", et "DTSTAMP" sont utilisées pour distinguer les divers usages de la méthode "REQUEST". Si la valeur de propriété "UID" dans la "REQUEST" n'est pas trouvée sur le calendrier du receveur, alors la "REQUEST" est pour une nouvelle chose à faire. Si la valeur de propriété "UID" est trouvée sur le calendrier du receveur, la "REQUEST" est une reprogrammation, une mise à jour, ou une reconfirmation de l'objet de calendrier "VTODO".

Si l'"Organisateur" de la méthode "REQUEST" n'est pas autorisé à faire une demande de chose à faire sur le système de calendrier du "Participant", une exception est alors retournée dans la propriété "REQUEST-STATUS" d'une méthode "REPLY" suivante, mais aucune action de programmation n'est effectuée.

Pour la méthode "REQUEST", plusieurs composants "VTODO" dans un seul objet iCalendar ne sont permis que pour les composants qui ont la même propriété "UID". C'est-à-dire, une série d'événements récurrents peut avoir des informations spécifiques de l'instance. Dans ce cas, plusieurs composants "VTODO" sont nécessaires pour exprimer la série entière.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:REQUEST d'un VTODO

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être REQUEST.
VTODO	1+	Tous les composants doivent avoir le même UID.
ATTENDEE	1+	
DTSTAMP	1	
DTSTART	1	
ORGANIZER	1	
PRIORITY	1	
SEQUENCE	0 ou 1	DOIT être présent si la valeur est supérieure à 0; PEUT être présent si 0.
SUMMARY	1	Peut être nul.
UID	1	
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	

COMMENT	0+	
COMPLETED	0 ou 1	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	Peut être nul.
DUE	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DUE NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PERCENT-COMPLETE	0 ou 1	
RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
RESOURCES	0+	
RRULE	0 ou 1	
STATUS	0 ou 1	PEUT être un de COMPLETED/NEEDS-ACTION/IN-PROCESS.
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
REQUEST-STATUS	0	
VALARM	0+	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VFREEBUSY	0	
VJOURNAL	0	

3.4.2.1 REQUEST pour reprogrammation d'un VTODO

La méthode "REQUEST" peut être utilisée pour reprogrammer un composant de calendrier "VTODO".

Reprogrammer un composant de calendrier "VTODO" implique un changement du composant de calendrier "VTODO" existant en termes d'heure de début ou d'heure prévue, d'intervalles de récurrence, et éventuellement de description. Si le CUA receveur d'une méthode "REQUEST" trouve que la valeur de propriété "UID" existe déjà dans le calendrier mais que la valeur de propriété "SEQUENCE" dans la "REQUEST" est supérieure à la valeur pour le "VTODO" existant, alors la méthode "REQUEST" décrit une reprogrammation du composant de calendrier "VTODO".

3.4.2.2 REQUEST pour mise à jour ou reconfirmation d'un VTODO

La méthode "REQUEST" peut être utilisée pour mettre à jour ou reconfirmer un composant de calendrier "VTODO". La reconfirmation est simplement une mise à jour de l'état d'achèvement de "Participant" ou de l'état global du composant de calendrier "VTODO".

Une mise à jour à un composant de calendrier "VTODO" existant n'implique pas de changement à l'heure de début ou à l'heure prévue, aux intervalles de récurrence, ou (généralement) à la description du composant de calendrier "VTODO". Si le CUA receveur d'une méthode "REQUEST" trouve que la valeur de propriété "UID" existe déjà dans le calendrier et que la valeur de propriété "SEQUENCE" dans la "REQUEST" est la même que la valeur pour l'événement existant, la méthode "REQUEST" décrit alors une mise à jour des détails du composant de calendrier "VTODO", mais pas une reprogrammation du composant de calendrier "VTODO".

La "REQUEST" de mise à jour est la réponse appropriée à une méthode "REFRESH" envoyée d'un "Participant" à l'"Organisateur" d'un composant de calendrier "VTODO".

Des méthodes "REQUEST" non sollicitées PEUVENT être envoyées par l'"Organisateur" d'un composant de calendrier "VTODO". Les méthodes "REQUEST" non sollicitées sont utilisées pour mettre à jour les détails du "VTODO" (sans le reprogrammer ou sans mettre à jour l'état d'achèvement des "Participants") ou le composant de calendrier "VTODO" lui-même (c'est-à-dire, reconfirmer le "VTODO").

3.4.2.3 REQUEST pour déléguer un VTODO

La méthode "REQUEST" est aussi utilisée pour déléguer ou réallouer la propriété d'un composant de calendrier "VTODO" à un autre "Utilisateur de calendrier". Par exemple, elle peut être utilisée pour déléguer un rôle de "Participant" (c'est-à-dire, "Président" ou "Participant") pour un composant de calendrier "VTODO". La méthode "REQUEST" est envoyée par un des "Participants" à un composant de calendrier "VTODO" existant à un autre individu.

Pour les besoins de cette description, le "Participant" qui délègue le composant de calendrier "VTODO" est appelé le "Déléguant". Le "participant" qui reçoit la demande de délégation est appelé le "Délégué".

Le "Déléguant" d'un composant de calendrier "VTODO" DOIT transmettre la méthode "REQUEST" existante pour un composant de calendrier "VTODO" au "Délégué". La description du composant de calendrier "VTODO" DOIT inclure la définition à jour du composant de calendrier "VTODO" du "Déléguant". La méthode "REQUEST" DOIT aussi inclure une propriété "ATTENDEE" avec l'adresse de calendrier du "Délégué". Le "Déléguant" DOIT aussi renvoyer une méthode "REPLY" à l'"Organisateur" avec la valeur du paramètre "PARTSTAT" de "Participant" du "Déléguant" réglée à "DELEGATED". De plus, le paramètre "DELEGATED-TO" DOIT être inclus avec l'adresse de calendrier du "Délégué". Une réponse à la "REQUEST" de délégation est envoyée du "Délégué" à l'"Organisateur", et facultativement au "Déléguant". La méthode "REPLY" provenant du "Délégué" DEVRAIT inclure la propriété "ATTENDEE" avec son adresse de calendrier et le paramètre "DELEGATED-FROM" avec la valeur de l'adresse de calendrier du "Déléguant".

La méthode "REQUEST" de délégation DOIT allouer une valeur pour le paramètre de propriété "RSVP" associé à la propriété "Participant" du "Déléguant" à cette propriété "ATTENDEE" du "Délégué". Par exemple, si la propriété "ATTENDEE" du "Déléguant" spécifie "RSVP=TRUE", alors la propriété "ATTENDEE" du "Délégué" DOIT spécifier "RSVP=TRUE".

3.4.2.4 REQUEST transmise à un utilisateur de calendrier non invité

Un "Participant" alloué à un composant de calendrier "VTODO" peut envoyer le composant de calendrier "VTODO" à un autre nouveau CU non associé précédemment au composant de calendrier "VTODO". Le "Participant" actuel alloué au composant de calendrier "VTODO" fait cela en transmettant la méthode "REQUEST" originale au nouveau CU. Le nouveau CU peut envoyer une "REPLY" à l'"Organisateur" du composant de calendrier "VTODO". La réponse contient une propriété "ATTENDEE" pour le nouveau CU.

L'"Organisateur" décide en fin de compte si le nouveau CU devient ou non partie à la chose à faire et n'est pas obligé de faire quelque chose avec une "REPLY" provenant du nouveau CU (non invité). Si l'"Organisateur" ne veut pas que le nouveau CU fasse partie de la chose à faire, la nouvelle propriété "ATTENDEE" n'est pas ajoutée au composant de calendrier "VTODO". L'"Organisateur" PEUT envoyer au CU un message "CANCEL" pour indiquer qu'il ne va pas être ajouté à la chose à faire. Si l'"Organisateur" décide d'ajouter le nouveau CU, la nouvelle propriété "ATTENDEE" est ajoutée au composant de calendrier "VTODO". De plus, l'"Organisateur" est libre de changer tout paramètre de propriété "ATTENDEE" à partir des valeurs fournies par le nouveau CU en quelque chose que l'"Organisateur" considère être approprié. L'"Organisateur" DEVRAIT envoyer au nouveau "Participant" un message "REQUEST" pour l'informer qu'il a été ajouté.

Quand il transmet une "REQUEST" à un autre CU, le "Participant" transmetteur NE DOIT PAS faire de changement au message original.

3.4.2.5 Demande de mise à jour d'état de participant

Un "Organisateur" d'un "VTODO" peut demander un état à jour à un ou plusieurs "Participants". L'"Organisateur" envoie une méthode "REQUEST" au "Participant" avec la séquence de propriété "ATTENDEE;RSVP=TRUE". La propriété "SEQUENCE" pour le "VTODO" n'est pas changée de sa valeur précédente. Un receveur détermine que le seul changement dans la "REQUEST" est que son paramètre de propriété "RSVP" indique une demande d'état à jour. Le receveur DEVRAIT répondre par une méthode "REPLY" indiquant son état actuel par rapport à la "REQUEST".

3.4.3 REPLY

La méthode "REPLY" dans un composant de calendrier "VTODO" est utilisée pour répondre (par exemple, accepter ou refuser) à une demande ou pour répondre à une demande de délégation. Elle est aussi utilisée par un "Participant" pour mettre à jour son état d'achèvement. Quand elle est utilisée pour fournir une réponse de délégation, le "Déléguant" DOIT inclure l'adresse de calendrier du "Délégué" dans le paramètre "DELEGATED-TO" de la propriété "ATTENDEE" du

"Déléguant". Le "Délégué" DOIT inclure l'adresse de calendrier du "Déléguant" dans le paramètre "DELEGATED-FROM" de la propriété "ATTENDEE" du "Délégué".

La méthode "REPLY" PEUT aussi être utilisée pour répondre à l'échec d'une méthode "REQUEST" de composant de calendrier "VTODO". Selon la valeur de "REQUEST-STATUS", aucune action de programmation peut n'avoir été effectuée.

L'"Organisateur" d'un composant de calendrier "VTODO" PEUT recevoir une méthode "REPLY" d'un "Utilisateur de calendrier" en dehors de la "REQUEST" originale. Par exemple, une méthode "REPLY" PEUT être reçue d'un "Délégué" d'un composant de calendrier "VTODO". De plus, la méthode "REPLY" PEUT être reçue d'un "Utilisateur de calendrier" inconnu à qui la "REQUEST" a été transmise par un "Participant" original du composant de calendrier "VTODO". Ce "Participant" non invité PEUT être accepté ou l'"organisateur" PEUT annuler le composant de calendrier "VTODO" pour le "Participant" non invité en lui envoyant une méthode "CANCEL".

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:REPLY d'un VTODO

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être REPLY.
VTODO	1+	Tous les composants DOIVENT avoir le même UID.
ATTENDEE	1	DOIT être l'adresse du participant qui répond.
DTSTAMP	1	
ORGANIZER	1	
REQUEST-STATUS	0+	
UID	1	DOIT être l'UID de la demande d'origine.
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
COMPLETED	0 ou 1	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	
DTSTART	0 ou 1	
DUE	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DUE NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PERCENT-COMPLETE	0 ou 1	
PRIORITY	0 ou 1	
RDATE	0+	
RELATED-TO	0+	
RESOURCES	0+	
RRULE	0 ou 1	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
SEQUENCE	0 ou 1	DOIT être le numéro de séquence de la REQUEST originale si supérieure à 0. PEUT être présent si 0.
STATUS	0 ou 1	
SUMMARY	0 ou 1	Peut être nul.
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
VALARM	0	
VTIMEZONE	0 ou 1	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VFREEBUSY	0	

3.4.4 ADD

La méthode "ADD" permet à l'"Organisateur" d'ajouter une ou plusieurs nouvelles instances à un "VTODO" existant en utilisant un seul message iTIP sans avoir à envoyer le "VTODO" entier avec toutes les données d'instance existantes, comme il l'aurait fait avec la méthode "REQUEST".

Le "UID" doit être celui de la chose à faire existante. Si la valeur de propriété "UID" dans le "ADD" n'est pas trouvée sur le calendrier du receveur, le receveur DEVRAIT alors envoyer un "REFRESH" à l'"Organisateur" afin qu'il soit mis à jour avec la dernière version du "VTODO". Si une mise en œuvre de "Participant" ne prend pas en charge la méthode "ADD", elle devrait répondre avec une valeur de "REQUEST-STATUS" de 3.14 et demander un "REFRESH".

Quand il traite un message "ADD", le "Participant" traite chaque composant du message "ADD" comme si il était référencé via un "RDATE" dans le composant principal.

La valeur de propriété "SEQUENCE" est incrémentée car la séquence de choses à faire a changé.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:ADD d'un VTODO

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être ADD.
VTODO	1	
DTSTAMP	1	
ORGANIZER	1	
PRIORITY	1	
SEQUENCE	1	DOIT être supérieure à 0.
SUMMARY	1	Peut être nul.
UID	1	DOIT correspondre à celui de la chose à faire originale.
ATTACH	0+	
ATTENDEE	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
COMPLETED	0 ou 1	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	Peut être nul.
DTSTART	0 ou 1	
DUE	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DUE NE DOIT PAS être présent.
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PERCENT-COMPLETE	0 ou 1	
RELATED-TO	0+	
RESOURCES	0+	
STATUS	0 ou 1	PEUT être un de COMPLETED/NEEDS-ACTION/IN-PROCESS.
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
EXDATE	0	
RECURRENCE-ID	0	
REQUEST-STATUS	0	
RDATE	0	
RRULE	0	
VALARM	0+	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	

VJOURNAL 0
VFREEBUSY 0

3.4.5 CANCEL

La méthode "CANCEL" dans un composant de calendrier "VTODO" est utilisée pour envoyer un avis d'annulation d'une demande de calendrier "VTODO" existante aux "Participants" affectés. Le message est envoyé par l'"Organisateur" d'un composant de calendrier "VTODO" aux "Participants" au composant de calendrier "VTODO". Pour un composant de calendrier "VTODO" récurrent, soit le composant de calendrier "VTODO" entier, soit des instances d'un composant de calendrier "VTODO" peuvent être annulés. Pour annuler la gamme complète d'un composant de calendrier "VTODO" récurrent, la valeur de propriété "UID" pour le composant de calendrier "VTODO" DOIT être spécifiée et un "RECURRENCE-ID" NE DOIT PAS être spécifié dans la méthode "CANCEL". Afin d'annuler une instance individuelle d'un composant de calendrier "VTODO" récurrent, la valeur de propriété "RECURRENCE-ID" pour le composant de calendrier "VTODO" DOIT être spécifiée dans la méthode "CANCEL".

Il y a deux options pour annuler une séquence d'instances d'un composant de calendrier "VTODO" récurrent :

- a. La propriété "RECURRENCE-ID" pour une instance dans la séquence DOIT être spécifiée avec la valeur "RANGE" de paramètre de propriété de "THISANDFUTURE" pour indiquer l'annulation du composant de calendrier "VTODO" spécifié et de toutes les instances qui suivent.
- b. Les instances individuelles de récurrence peuvent être annulées en spécifiant plusieurs composants "VTODO" chacun avec une propriété "RECURRENCE-ID" correspondant à une des instances à annuler.

L'"Organisateur" DOIT envoyer un message "CANCEL" à chaque "Participant" affecté par l'annulation. Ce peut être fait en utilisant un seul message "CANCEL" pour tous les "Participants" ou plusieurs messages avec différents sous ensembles des "Participants" affectés dans chaque.

Quand a "VTODO" est annulé, la valeur de propriété "SEQUENCE" DOIT être incrémentée comme décrit au paragraphe 2.1.4.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:CANCEL d'un VTODO

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être CANCEL.
VTODO	1+	
ATTENDEE	0+	DOIT inclure certains ou tous les participants retirés de la chose à faire. DOIT inclure certains ou tous les participants si la chose à faire entière est annulée.
UID	1	DOIT faire écho à l'UID original.
DTSTAMP	1	
ORGANIZER	1	
SEQUENCE	1	
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
COMPLETED	0 ou 1	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	
DTSTART	0 ou 1	
DUE	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DUE NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PERCENT-COMPLETE	0 ou 1	
RDATE	0+	

RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance de composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
RESOURCES	0+	
RRULE	0 ou 1	
PRIORITY	0 ou 1	
STATUS	0 ou 1	DOIT être réglé à CANCELLED pour annuler le VTODO entier. NE DOIT PAS être inclus si on retire des participants spécifiques.
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
REQUEST-STATUS	0	
VALARM	0	
VTIMEZONE	0 ou 1	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VFREEBUSY	0	

3.4.6 REFRESH

La méthode "REFRESH" dans un composant de calendrier "VTODO" est utilisée par les "Participants" à un composant de calendrier "VTODO" existant pour demander à l'"Organisateur" une description à jour du composant de calendrier "VTODO". L'"Organisateur" du composant de calendrier "VTODO" PEUT utiliser cette méthode pour demander un état à jour aux "Participants". La méthode "REFRESH" DOIT spécifier la propriété "UID" correspondant au composant de calendrier "VTODO" qui a besoin de la mise à jour.

Un rafraîchissement d'une instance récurrente de composant de calendrier "VTODO" peut être demandée en spécifiant la propriété "RECURRENCE-ID" correspondant au composant de calendrier "VTODO" associé. L'"Organisateur" répond avec la dernière description et le dernier rendu du composant de calendrier "VTODO". Dans la plupart des cas, cela va être une "REQUEST" sauf si le "VTODO" a été annulé, et dans ce cas, l'"Organisateur" DOIT envoyer un "CANCEL". Cette méthode est destinée à faciliter le traitement par la machine des demandes de mise à jour d'un composant de calendrier "VTODO".

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:REFRESH d'un VTODO

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être REFRESH.
VTODO	1	
ATTENDEE	1	
DTSTAMP	1	
UID	1	DOIT faire écho à l'UID original.
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
IANA-PROPERTY	0+	
X-PROPERTY	0+	
ATTACH	0	
CATEGORIES	0	
CLASS	0	
COMMENT	0	
COMPLETED	0	
CONTACT	0	
CREATED	0	
DESCRIPTION	0	
DTSTART	0	
DUE	0	
DURATION	0	
EXDATE	0	
GEO	0	
LAST-MODIFIED	0	

LOCATION	0
ORGANIZER	0
PERCENT-COMPLETE	0
PRIORITY	0
RDATE	0
RELATED-TO	0
REQUEST-STATUS	0
RESOURCES	0
RRULE	0
SEQUENCE	0
STATUS	0
URL	0
VALARM	0
VTIMEZONE	0+
IANA-COMPONENT	0+
X-COMPONENT	0+
VEVENT	0
VFREEBUSY	0

3.4.7 COUNTER

La méthode "COUNTER" dans un composant de calendrier "VTODO" est utilisée par un "Participant" d'un composant de calendrier "VTODO" existant pour soumettre à l'"Organisateur" une contre proposition pour le composant de calendrier "VTODO".

La contre proposition est un objet iCalendar consistant en un composant de calendrier "VTODO" qui fournit la description complète du composant de calendrier "VTODO" de remplacement.

L'"Organisateur" rejette la contre proposition en envoyant au "Participant" une méthode "DECLINECOUNTER". L'"Organisateur" accepte la contre proposition en reprogrammant la chose à faire comme décrit au paragraphe 3.4.2.1, "REQUEST pour reprogrammer une chose à faire". Le CUA de l'"Organisateur" DEVRAIT envoyer un message "REQUEST" à tous les "Participants" affectés par tout changement déclenché par une "COUNTER" acceptée.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:COUNTER d'un VTODO

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être COUNTER.
VTODO	1	
ATTENDEE	1+	
DTSTAMP	1	
ORGANIZER	1	
PRIORITY	1	
SUMMARY	1	Peut être nul.
UID	1	
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
COMPLETED	0 ou 1	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	Peut être nul.
DTSTART	0 ou 1	
DUE	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DUE NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PERCENT-COMPLETE	0 ou 1	

RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
REQUEST-STATUS	0+	
RESOURCES	0+	
RRULE	0 ou 1	
SEQUENCE	0 ou 1	DOIT faire écho au numéro de séquence original. DOIT être présent si non zéro. PEUT être présent si zéro.
STATUS	0 ou 1	PEUT être un de COMPLETED/NEEDS-ACTION/IN-PROCESS/CANCELLED.
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
VALARM	0+	
VTIMEZONE	0 ou 1	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VFREEBUSY	0	

3.4.8 DECLINECOUNTER

La méthode "DECLINECOUNTER" dans un composant de calendrier "VTODO" est utilisée par un "Organisateur" du composant de calendrier "VTODO" pour rejeter une contre proposition offerte par un des "Participants". L'"Organisateur" envoie le message au "Participant" qui lui a envoyé la méthode "COUNTER".

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:DECLINECOUNTER d'un VTODO

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être DECLINECOUNTER.
VTODO	1	
ATTENDEE	1+	DOIT être pour tous les participants.
DTSTAMP	1	
ORGANIZER	1	
SEQUENCE	1	DOIT faire écho du numéro de séquence d'origine.
UID	1	DOIT faire écho de l'UID original.
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
COMPLETED	0 ou 1	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	
DTSTART	0 ou 1	
DUE	0 ou 1	Si présent, DURATION NE DOIT PAS être présent.
DURATION	0 ou 1	Si présent, DUE NE DOIT PAS être présent.
EXDATE	0+	
GEO	0 ou 1	
LAST-MODIFIED	0 ou 1	
LOCATION	0 ou 1	
PERCENT-COMPLETE	0 ou 1	
PRIORITY	0 ou 1	
RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance d'un composant de calendrier récurrent. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
REQUEST-STATUS	0+	
RESOURCES	0+	
RRULE	0 ou 1	

STATUS	0 ou 1	PEUT être un de COMPLETED/NEEDS-ACTION/IN-PROCESS.
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
VALARM	0	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VFREEBUSY	0	

3.5 Méthodes pour composants VJOURNAL

Ce paragraphe définit les propriétés des méthodes applicables au composant de calendrier "VJOURNAL".

On résume ci-dessous les méthodes définies pour le composant de calendrier "VJOURNAL".

Méthode Description

PUBLISH	Envoie une entrée de journal. Utilisé principalement comme méthode d'annonce de l'existence d'une entrée de journal.
ADD	Ajoute une ou plusieurs instances à une entrée de journal existante.
CANCEL	Annule une ou plusieurs instances d'une entrée de journal existante.

3.5.1 PUBLISH

La méthode "PUBLISH" dans un composant de calendrier "VJOURNAL" n'a pas de réponse associée. Elle est simplement l'envoi d'un objet iCalendar qui peut être ajouté à un calendrier. Elle DOIT avoir un "Organisateur". Elle NE DOIT PAS avoir de "Participant". L'usage prévu est d'encapsuler une entrée de journal arbitraire comme objet iCalendar. L'"Organisateur" PEUT ensuite mettre à jour (avec une autre méthode "PUBLISH") ou annuler (avec une méthode "CANCEL") une entrée de journal précédemment publiée.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:PUBLISH d'un VJOURNAL

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être PUBLISH.
VJOURNAL	1+	
DESCRIPTION	1	Peut être nul.
DTSTAMP	1	
DTSTART	1	
ORGANIZER	1	
UID	1	
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
CONTACT	0+	
CREATED	0 ou 1	
EXDATE	0+	
LAST-MODIFIED	0 ou 1	
RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance récurrente de composant de calendrier. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
RRULE	0 ou 1	
SEQUENCE	0 ou 1	DOIT être présent si non zéro. PEUT être présent si zéro.
STATUS	0 ou 1	PEUT être un de DRAFT/FINAL/CANCELLED.
SUMMARY	0 ou 1	Peut être nul.
URL	0 ou 1	
IANA-PROPERTY	0+	

X-PROPERTY	0+	
ATTENDEE	0	
REQUEST-STATUS	0	
VALARM	0+	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VFREEBUSY	0	
VTODO	0	

3.5.2 ADD

La méthode "ADD" permet à l'"organisateur" d'ajouter une ou plusieurs nouvelles instances à un "VJOURNAL" existant en utilisant un seul message iTIP sans avoir à envoyer le "VJOURNAL" entier avec toutes les données de l'instance existante, comme il aurait fait si il avait utilisé la méthode "REQUEST".

L'"UID" doit être celui de l'entrée de journal existante. Si la valeur de propriété "UID" dans le "ADD" n'est pas trouvée dans le calendrier du receveur, alors le receveur PEUT traiter le "ADD" comme une méthode "PUBLISH".

Lors du traitement d'un message "ADD", le "participant" traite chaque composant dans le message "ADD" comme si il était référencé via un "RDATE" dans le composant principal. Il n'y a pas de réponse à l'"organisateur".

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:ADD d'un VJOURNAL

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être ADD.
VJOURNAL	1	
DESCRIPTION	1	Peu être nul.
DTSTAMP	1	
DTSTART	1	
ORGANIZER	1	
SEQUENCE	1	DOIT être supérieure à 0.
UID	1	DOIT correspondre à celui du journal original.
ATTACH	0+	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
CONTACT	0+	
CREATED	0 ou 1	
LAST-MODIFIED	0 ou 1	
RELATED-TO	0+	
STATUS	0 ou 1	PEUT être un de DRAFT/FINAL/CANCELLED.
SUMMARY	0 ou 1	Peut être nul.
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
ATTENDEE	0	
EXDATE	0	
RECURRENCE-ID	0	
REQUEST-STATUS	0	
RDATE	0	
RRULE	0	
VALARM	0+	
VTIMEZONE	0 ou 1	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VFREEBUSY	0	
VTODO	0	

3.5.3 CANCEL

La méthode "CANCEL" dans un composant de calendrier "VJOURNAL" est utilisée pour envoyer un avis d'annulation d'une entrée de journal existante. Le message est envoyé par l'"organisateur" d'une entrée de journal. Pour une entrée de journal récurrente, soit l'entrée de journal entière, soit des instances d'une entrée de journal, peuvent être annulées. Pour annuler la gamme complète d'une entrée de journal récurrente, la valeur de propriété "UID" pour l'entrée de journal DOIT être spécifiée et une propriété "RECURRENCE-ID" NE DOIT PAS être spécifiée dans la méthode "CANCEL". Afin d'annuler une instance individuelle de l'entrée de journal, la valeur de propriété "RECURRENCE-ID" pour l'entrée de journal DOIT être spécifiée dans la méthode "CANCEL".

Il y a deux options pour annuler une séquence d'instances d'un composant de calendrier "VJOURNAL" récurrent :

- La propriété "RECURRENCE-ID" pour une instance dans la séquence DOIT être spécifiée avec la valeur de paramètre de propriété "RANGE" de "THISANDFUTURE" pour indiquer l'annulation du composant de calendrier "VJOURNAL" spécifié et de toutes les instances après lui.
- Des instances de récurrence individuelles peuvent être annulées en spécifiant plusieurs composants "VJOURNAL" avec chacun une propriété "RECURRENCE-ID" correspondant à une des instances à annuler.

Quand un "VJOURNAL" est annulé, la valeur de propriété "SEQUENCE" DOIT être incrémentée comme décrit au paragraphe 2.1.4.

Ce type de méthode est un objet iCalendar qui se conforme aux contraintes de propriété suivantes :

Contraintes pour une METHOD:CANCEL d'un VJOURNAL

Composant/propriété	Présence	commentaire
METHOD	1	DOIT être CANCEL.
VJOURNAL	1+	Tous DOIVENT avoir le même UID.
DTSTAMP	1	
ORGANIZER	1	
SEQUENCE	1	
UID	1	DOIT être l'UID de la demande originale.
ATTACH	0+	
ATTENDEE	0	
CATEGORIES	0+	
CLASS	0 ou 1	
COMMENT	0+	
CONTACT	0+	
CREATED	0 ou 1	
DESCRIPTION	0 ou 1	
DTSTART	0 ou 1	
EXDATE	0+	
LAST-MODIFIED	0 ou 1	
RDATE	0+	
RECURRENCE-ID	0 ou 1	Seulement si il se réfère à une instance récurrente de composant de calendrier. Autrement, il NE DOIT PAS être présent.
RELATED-TO	0+	
RRULE	0 ou 1	
STATUS	0 ou 1	PEUT être présent; DOIT être CANCELLED si présent.
SUMMARY	0 ou 1	
URL	0 ou 1	
IANA-PROPERTY	0+	
X-PROPERTY	0+	
REQUEST-STATUS	0	
VALARM	0	
VTIMEZONE	0+	DOIT être présent si une date/heure se réfère à une zone horaire.
IANA-COMPONENT	0+	
X-COMPONENT	0+	
VEVENT	0	
VFREEBUSY	0	
VTODO	0	

3.6 Réponses d'état

La propriété "REQUEST-STATUS" est utilisée pour porter les informations d'état sur un message iTIP "REPLY", "COUNTER", ou "DECLINECOUNTER". Les codes qui figurent dans le tableau ci-dessous DEVRAIENT être utilisés. Si la propriété "REQUEST-STATUS" n'est pas présente dans un de ces messages iTIP, un code d'état de "2.0" (succès) DOIT alors être supposé.

La présente spécification ajoute un nouveau registre IANA pour les valeurs de propriété "REQUEST-STATUS", comme défini à la Section 7, qui comporte un nouveau gabarit d'enregistrement pour définir les composants spécifiques de la valeur de propriété "REQUEST-STATUS". Des codes supplémentaires PEUVENT être utilisés, pourvu que le processus décrit au paragraphe 8.2.1 de la [RFC5545] soit utilisé pour les enregistrer.

La présente spécification permet que plusieurs propriétés "REQUEST-STATUS" soient retournées dans les composants iCalendar des messages iTIP appropriés. Quand plusieurs propriétés "REQUEST-STATUS" sont présentes, les restrictions suivantes s'appliquent :

1. Au sein de tout composant, la valeur numérique de "niveau supérieur" du "code court d'état de retour" DOIT être la même pour toutes les propriétés "REQUEST-STATUS", c'est-à-dire, il ne peut pas y avoir de mélange, par exemple, de codes 2.xx et 5.xx au sein d'un seul composant.
2. Sur tous les composants du message iTIP, ceci s'(applique :
 - A. Si un composant a le code 5.xx, alors soit tous les composants DOIVENT avoir un code dans cette gamme, soit "REQUEST-STATUS" NE DOIT PAS être présent dans les autres composants si un code 5.xx n'est pas approprié pour ces composants.
 - B. Autrement, si un composant devrait avoir un code 3.xx, alors soit tous les composants DOIVENT avoir un a code dans cette gamme, soit "REQUEST-STATUS" NE DOIT PAS être présent dans les autres composants si un code 3.xx n'est pas approprié pour ces composants.
 - C. Les codes 2.xx et 4.xx peuvent être utilisés dans des composants différents, pourvu que chaque composant suive la restriction du (1) ci-dessus.

Les codes de "REQUEST-STATUS" suivants sont définis (toutes les "données offensantes" PEUVENT être spécifiées dans la valeur "REQUEST-STATUS" comme champ extdata) :

3.6.1 Code d'état 2.0

Description d'état : succès.

Données d'exception d'état : aucune.

Description : l'opération iTIP a réussi.

3.6.2 Code d'état 2.1

Description d'état : succès, mais repli effectué sur une ou plusieurs valeurs de propriété.

Données d'exception d'état : le nom et la valeur de la propriété PEUVENT être spécifiés.

Description : l'opération iTIP a réussi avec un repli sur une ou plusieurs des valeurs de propriété.

3.6.3 Code d'état 2.2

Description d'état : succès ; propriété invalide ignorée.

Données d'exception d'état : le nom de propriété PEUT être spécifié.

Description : l'opération iTIP a réussi mais une propriété a été ignorée.

3.6.4 Code d'état 2.3

Description d'état : succès ; paramètre de propriété invalide ignoré.

Données d'exception d'état : le nom et la valeur du paramètre de propriété PEUVENT être spécifiés.

Description : l'opération iTIP a réussi mais un paramètre de propriété a été ignoré parce qu'invalide.

3.6.5 Code d'état 2.4

Description d'état : succès ; propriété inconnue, non standard ignorée.

Données d'exception d'état : le nom de la propriété non standard PEUT être spécifié.

Description : l'opération iTIP a réussi mais un paramètre de propriété a été ignoré parce qu'inconnu.

3.6.6 Code d'état 2.5

Description d'état : succès ; valeur de propriété inconnue, non standard, ignorée.

Données d'exception d'état : la valeur de propriété et non standard PEUT être spécifiée.

Description : l'opération iTIP a réussi mais une propriété a été ignorée parce que sa valeur est inconnue.

3.6.7 Code d'état 2.6

Description d'état : succès ; composant de calendrier invalide ignoré.

Données d'exception d'état : un composant de calendrier sentinelle (par exemple, BEGIN: ALARM) PEUT être spécifié.

Description : l'opération iTIP a réussi mais un composant a été ignoré parce qu'invalide.

3.6.8 Code d'état 2.7

Description d'état : succès ; demande transmise à l'utilisateur de calendrier.

Données d'exception d'état : les adresses d'utilisateur de calendrier d'origine et transmise PEUVENT être spécifiées.

Description : l'opération iTIP a réussi, et la demande a été transmise à un autre Utilisateur de calendrier.

3.6.9 Code d'état 2.8

Description d'état : succès ; l'événement répété est ignoré. Programmé comme un seul composant.

Données d'exception d'état : le nom et la valeur de propriété RRULE ou RDATE PEUVENT être spécifiés.

Description : l'opération iTIP a réussi mais un événement répété a été tronqué à une seule instance.

3.6.10 Code d'état 2.9

Description d'état : succès ; date et heure de fin tronquée à la limite de date.

Données d'exception d'état : la valeur de propriété DTEND PEUT être spécifiée.

Description : l'opération iTIP a réussi mais l'heure de fin a été tronquée à la limite de date.

3.6.11 Code d'état 2.10

Description d'état : succès ; VTODO répété ignoré. Programmé comme un seul VTODO.

Données d'exception d'état : le nom ou la valeur de la propriété RRULE ou RDATE PEUVENT être spécifiés.

Description : l'opération iTIP a réussi mais une chose à faire répétée a été tronquée en une seule instance.

3.6.12 Code d'état 2.11

Description d'état : succès ; une RRULE sans limite a été coupée à un nombre fini d'instances.

Données d'exception d'état : le nom et la valeur de la propriété RRULE PEUVENT être spécifiés. Le nombre d'instances PEUT aussi être spécifié.

Description : l'opération iTIP a réussi mais un objet répété sans limite a été coupé à un nombre d'instances fini.

3.6.13 Code d'état 3.0

Description d'état : nom de propriété invalide.

Données d'exception d'état : le nom de propriété PEUT être spécifié.

Description : l'opération iTIP a échoué à cause d'un nom de propriété invalide.

3.6.14 Code d'état 3.1

Description d'état : valeur de propriété invalide.

Données d'exception d'état : le nom et la valeur de la propriété PEUVENT être spécifiés.

Description : l'opération iTIP a échoué à cause d'une valeur de propriété invalide.

3.6.15 Code d'état 3.2

Description d'état : paramètre de propriété invalide.

Données d'exception d'état : le nom et la valeur du paramètre de la propriété PEUVENT être spécifiés.

Description : l'opération iTIP a échoué à cause d'un paramètre de propriété invalide.

3.6.16 Code d'état 3.3

Description d'état : valeur de paramètre de propriété invalide.

Données d'exception d'état : le nom et la valeur du paramètre de la propriété PEUVENT être spécifiés.

Description : l'opération iTIP a échoué à cause d'une valeur invalide de paramètre de propriété.

3.6.17 Code d'état 3.4

Description d'état : séquence invalide de composants de calendrier.

Données d'exception d'état : un composant de calendrier sentinelle PEUT être spécifié (par exemple, BEGIN:VTIMEZONE).

Description : l'opération iTIP a échoué à cause d'un composant invalide.

3.6.18 Code d'état 3.5

Description d'état : date ou heure invalide.

Données d'exception d'état : la ou les valeurs de date/heure PEUVENT être spécifiées.

Description : l'opération iTIP a échoué à cause d'une propriété invalide de date ou heure.

3.6.19 Code d'état 3.6

Description d'état : règle invalide.

Données d'exception d'état : la valeur de propriété RRULE PEUT être spécifiée.

Description : l'opération iTIP a échoué à cause d'une propriété de règle invalide.

3.6.20 Code d'état 3.7

Description d'état : utilisateur de calendrier invalide.

Données d'exception d'état : la valeur de propriété ATTENDEE PEUT être spécifiée.

Description : l'opération iTIP a échoué à cause d'une propriété ATTENDEE invalide.

3.6.21 Code d'état 3.8

Description d'état : non autorisé.

Données d'exception d'état : les valeurs de propriété METHOD et ATTENDEE PEUVENT être spécifiées.

Description : l'opération iTIP a échoué à cause d'un participant qui n'a pas les privilèges convenables pour l'opération.

3.6.22 Code d'état 3.9

Description d'état : version non prise en charge.

Données d'exception d'état : le nom et la valeur de la propriété VERSION PEUVENT être spécifiés.

Description : l'opération iTIP a échoué parce que la version des données de calendrier n'est pas prise en charge.

3.6.23 Code d'état 3.10

Description d'état : l'entité de la demande est trop grande.

Données d'exception d'état : aucune.

Description : l'opération iTIP a échoué parce que les données de calendrier sont trop grandes.

3.6.24 Code d'état 3.11

Description d'état : absence d'un composant ou propriété exigé.

Données d'exception d'état : le nom de composant ou propriété PEUT être spécifié.

Description : l'opération iTIP a échoué parce que les données de calendrier ne contiennent pas une propriété ou composant exigé.

3.6.25 Code d'état 3.12

Description d'état : un composant ou propriété inconnu a été trouvé.

Données d'exception d'état : le nom du composant ou propriété PEUT être spécifié.

Description : l'opération iTIP a échoué parce que les données de calendrier contiennent une propriété ou composant inconnu.

3.6.26 Code d'état 3.13

Description d'état : un composant ou propriété non pris en charge a été trouvé.

Données d'exception d'état : le nom du composant ou propriété PEUT être spécifié.

Description : l'opération iTIP a échoué parce que les données de calendrier contiennent une propriété ou composant non pris en charge.

3.6.27 Code d'état 3.14

Description d'état : capacité non prise en charge.

Données d'exception d'état : la méthode ou action PEUT être spécifiée.

Description : l'opération iTIP a échoué parce qu'elle n'est pas prise en charge.

3.6.28 Code d'état 4.0

Description d'état : conflit d'événements. La date/heure est occupée.

Données d'exception d'état : les noms et valeurs de propriété DTSTART et DTEND PEUVENT être spécifiés.

Description : l'opération iTIP a échoué parce que l'événement chevauche la date et l'heure d'un autre événement.

3.6.29 Code d'état 5.0

Description d'état : demande non prise en charge.

Données d'exception d'état : la valeur de propriété METHOD PEUT être spécifiée.

Description : l'opération iTIP a échoué parce que l'opération n'est pas prise en charge.

3.6.30 Code d'état 5.1

Description d'état : Service indisponible.

Données d'exception d'état : la valeur de propriété ATTENDEE PEUT être spécifiée.

Description : l'opération iTIP a échoué parce que la programmation n'est pas activée.

3.6.31 Code d'état 5.2

Description d'état : service de calendrier invalide.

Données d'exception d'état : la valeur de propriété ATTENDEE PEUT être spécifiée.

Description : l'opération iTIP a échoué parce que il n'y a pas de capacité de programmation.

3.6.32 Code d'état 5.3

Description d'état : l'utilisateur ne prend pas en charge la programmation.

Données d'exception d'état : la valeur de propriété ATTENDEE PEUT être spécifiée.

Description : l'opération iTIP a échoué parce que la programmation n'est pas activée pour un participant.

3.7 Considérations de mise en œuvre**3.7.1 Travail avec des instances de récurrence**

iCalendar inclut une grammaire de récurrence pour représenter les événements récurrents. L'avantage d'une telle grammaire est la capacité de représenter un certain nombre d'événements dans un seul objet. Cependant, bien que cela simplifie la création d'un événement récurrent, les instances qui y satisfont doivent quand même être référencées. Par exemple, un "Participant" peut refuser la troisième instance d'un événement récurrent les vendredis. De même, l'"Organisateur" peut changer l'heure ou le lieu d'une seule instance de l'événement récurrent.

Comme les mises en œuvre peuvent choisir de mémoriser les événements récurrents comme un seul objet ou comme une collection d'objets d'événement discrets, en rapports, le protocole est conçu de façon à ce que chaque instance récurrente puisse être à la fois référencée et munie d'un numéro de version. Donc, les mises en œuvre qui choisissent de conserver les propriétés par instance (comme le paramètre "PARTSTAT" de la propriété "ATTENDEE") peuvent le faire. Cependant, le protocole n'exige pas une reconnaissance par instance sauf si l'instance elle-même doit être renégoziée.

La liste des scénarios pour la référence d'instance récurrente figure ci-dessous. Dans un but de simplification, un changement à un événement se réfère à une "propriété déclencheuse". C'est-à-dire, une propriété qui a un effet substantiel sur la réunion elle-même, comme l'heure de début, le lieu, la date prévue (pour les composants de calendrier "VTODO") et

éventuellement la description.

Action initiés par l'"organisateur" :

- o supprime ou change une seule instance d'un événement récurrent,
- o fait des changements qui affectent toutes les instances futures,
- o fait des changements qui affectent toutes les instances précédentes,
- o supprime ou modifie une instance changée précédemment.

Action initiées par un "participant" :

- o change l'état pour une instance récurrente particulière,
- o envoie un "COUNTER" pour une instance récurrente particulière.

Une instance d'un événement récurrent reçoit une identification univoque, la propriété "RECURRENCE-ID", quand cette instance est renégociée. La négociation peut être nécessaire quand un changement substantiel de l'événement ou de la chose à faire a été effectué (comme de changer l'heure de début, de fin, la date, ou le lieu). L'"Organisateur" peut identifier une instance récurrente spécifique en utilisant la propriété "RECURRENCE-ID". La valeur de propriété est égale à la date/heure de l'instance. Si l'"Organisateur" souhaite changer le "DTSTART", la valeur originale, non modifiée de "DTSTART" de l'instance est utilisée comme valeur de la propriété "RECURRENCE-ID", et les nouvelles valeurs de "DTSTART" et "DTEND" reflètent le changement.

3.7.2 Considérations de propriété de participant

La propriété "ORGANIZER" est exigée sur les événements publiés, les choses à faire, et les entrées de journal pour deux raisons. D'abord, seul l'"Organisateur" a la permission de mettre à jour et redistribuer un composant d'événement ou de chose à faire. Il s'ensuit que la propriété "ORGANIZER" DOIT être présente dans le composant d'événement, de chose à faire, ou d'entrée de journal afin que le CUA ait une base pour autoriser une mise à jour. Ensuite, il est prudent de fournir un point de contact pour quiconque reçoit un composant publié, en cas de problèmes.

Les adresses de messagerie électronique qui correspondent à des groupes d'"Utilisateurs de calendrier" pourraient être spécifiées comme un URI mailto: [RFC2368] d'adresse d'utilisateur de calendrier. Envoyer un message électronique à une telle adresse résulte en l'envoi du message à plusieurs receveurs. Une telle adresse peut être utilisée comme valeur d'une propriété "ATTENDEE". Donc, il est possible que le receveur d'une "REQUEST" n'apparaisse pas explicitement dans la liste.

Il est recommandé que l'approche générale pour trouver un "Utilisateur de calendrier" dans une liste de "participant" soit comme suit :

1. Chercher l'"Utilisateur de calendrier" dans la liste des "Participants" où "CUTYPE=INDIVIDUAL".
2. En cas d'échec de (1), chercher des "Participants" où "CUTYPE=GROUP" ou "CUTYPE=UNKNOWN". Le CUA détermine alors si l'"Utilisateur de calendrier" est membre d'un de ces groupes. Si il l'est, la méthode "REPLY" envoyée à l'"Organisateur" DOIT contenir une nouvelle propriété "ATTENDEE" dans laquelle :
 - * le paramètre de propriété "TYPE" est réglé à INDIVIDUAL
 - * le paramètre de propriété "MEMBER" est réglé au nom du groupe.
3. En cas d'échec de (2), le CUA PEUT ignorer ou accepter la demande au choix de l'"Utilisateur de calendrier".

3.7.3 Jetons d'extension

Pour rendre les objets iCalendar extensibles, de nouveaux composants, propriétés, ou paramètres de propriété peuvent être utilisés. Deux types d'extensions sont définis dans la [RFC5545] : les jetons enregistrés par l'IANA ("iana-token") et les jetons d'utilisation expérimentale ("x-name"). Un client DEVRAIT sauvegarder les "iana-token" et PEUT les utiliser dans les réponses. Un client PEUT sauvegarder les "x-name" et PEUT les utiliser dans les réponses. Quand il délègue ou transmet des messages aux autres CU, un client DEVRAIT inclure les "iana-token" et "x-names".

4. Exemples

4.1 Exemples d'événement publié

Dans le contexte de calendrier et programmation, la publication se réfère au transfert unidirectionnel des informations

d'événement. Les consommateurs d'événements publiés incorporent simplement l'événement dans un calendrier. Aucune réponse n'est attendue. L'individu "A" publie un événement. L'individu "B" lit l'événement et l'incorpore dans son calendrier. Les événements sont publiés de plusieurs façons, incluant d'incorporer l'événement comme objet dans une page de la Toile, de communiquer l'événement sur un message à une liste de diffusion, ou d'envoyer l'événement à un groupe de nouvelles.

Le tableau ci-dessous illustre la séquence d'événements entre le publieur et les consommateurs d'un événement publié.

Action	Organisateur	Receveur
Publier un événement	"A" envoie ou affiche un message PUBLISH.	"B" lit un événement publié.
Publier un événement mis à jour	"A" envoie ou affiche un message PUBLISH.	"B" lit l'événement mis à jour.
Annuler un événement publié	"A" envoie ou affiche un message CANCEL.	"B" lit la publication d'annulation de l'événement.

4.1.1 Événement publié minimal

L'objet iCalendar ci-dessous décrit un seul événement qui commence le 1er juillet 1997 à 20:00 UTC. Cet événement contient l'ensemble minimum de propriétés pour un "PUBLISH" pour un composant de calendrier "VEVENT".

```
BEGIN:VCALENDAR
METHOD:PUBLISH
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
DTSTART:19970701T200000Z
DTSTAMP:19970611T190000Z
SUMMARY:ST. PAUL SAINTS -VS- DULUTH-SUPERIOR DUKES
UID:0981234-1234234-23@exemple.com
END:VEVENT
END:VCALENDAR
```

4.1.2 Changement d'un événement publié

L'objet iCalendar ci-dessous décrit une mise à jour de l'événement décrit au paragraphe 4.1.1 ; l'heure a été changée, une heure de fin a été ajoutée, et le numéro de séquence a été ajusté.

```
BEGIN:VCALENDAR
METHOD:PUBLISH
VERSION:2.0
PRODID:-//Example/ExampleCalendarClient//EN
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
DTSTAMP:19970612T190000Z
DTSTART:19970701T210000Z
DTEND:19970701T230000Z
SEQUENCE:1
UID:0981234-1234234-23@exemple.com
SUMMARY:ST. PAUL SAINTS -VS- DULUTH-SUPERIOR DUKES
END:VEVENT
END:VCALENDAR
```

La propriété "UID" est utilisée par le client pour identifier l'événement. La propriété "SEQUENCE" indique que c'est un changement à l'événement. L'événement avec un "UID" correspondant et le numéro de séquence 0 est remplacé par cet événement.

La propriété "SEQUENCE" donne un moyen fiable pour distinguer différentes versions du même événement. Chaque fois qu'un événement est publié, son numéro de séquence est incrémenté. Si un client reçoit un événement avec un numéro de séquence 5 et trouve qu'il a le même événement avec le numéro de séquence 2, l'événement DEVRAIT être mis à jour. Cependant, si le client a reçu un événement avec le numéro de séquence 2 et trouve qu'il a déjà le numéro de séquence 5 du même événement, l'événement NE DOIT PAS être mis à jour.

4.1.3 Annulation d'un événement publié

L'objet iCalendar ci-dessous annule l'événement décrit au paragraphe 4.1.1. Cela annule l'événement avec la propriété "SEQUENCE" de 0, 1, et 2.

```
BEGIN:VCALENDAR
METHOD:CANCEL
VERSION:2.0
PRODID:-//Example/ExampleCalendarClient//EN
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
COMMENT:DUKES forfait the game
SEQUENCE:2
UID:0981234-1234234-23@exemple.com
DTSTAMP:19970613T190000Z
END:VEVENT
END:VCALENDAR
```

4.1.4 Riche événement publié

Cet exemple décrit le même événement qu'au paragraphe 4.1.1, mais en plus grand détail.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:PUBLISH
SCALE:GREGORIAN
VERSION:2.0
BEGIN:VTIMEZONE
TZID:America-Chicago
TZURL:http://exemple.com/tz/America-Chicago
BEGIN:STANDARD
DTSTART:19671029T020000
RRULE:FREQ=YEARLY;BYDAY=-1SU;BYMONTH=10
TZOFFSETFROM:-0500
TZOFFSETTO:-0600
TZNAME:CST
END:STANDARD
BEGIN:DAYLIGHT
DTSTART:19870405T020000
RRULE:FREQ=YEARLY;BYDAY=1SU;BYMONTH=4
TZOFFSETFROM:-0600
TZOFFSETTO:-0500
TZNAME:CDT
END:DAYLIGHT
END:VTIMEZONE
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTACH:http://www.exemple.com/
CATEGORIES:SPORTS EVENT,ENTERTAINMENT
CLASS:PRIVATE
DESCRIPTION:MIDWAY STADIUM\n
Big time game. MUST see.\n
Expected duration:2 hours\n
DTEND;TZID=America-Chicago:19970701T180000
DTSTART;TZID=America-Chicago:19970702T160000
DTSTAMP:19970614T190000Z
STATUS:CONFIRMED
LOCATION;VALUE=URI:http://stadium.exemple.com/
PRIORITY:2
RESOURCES:SCOREBOARD
```

```

SEQUENCE:3
SUMMARY:ST. PAUL SAINTS -VS- DULUTH-SUPERIOR DUKES
UID:0981234-1234234-23@exemple.com
RELATED-TO:0981234-1234234-14@exemple.com
BEGIN:VALARM
TRIGGER:-PT2H
ACTION:DISPLAY
DESCRIPTION: Il faut maintenant partir pour le stade.
END:VALARM
BEGIN:VALARM
TRIGGER:-PT30M
ACTION:AUDIO
END:VALARM
END:VEVENT
END:VCALENDAR

```

Le champ "RELATED-TO" contient la propriété "UID" d'un événement de calendrier en rapport. La propriété "SEQUENCE" 3 indique que cet événement remplace les versions 0, 1, et 2.

4.1.5 Anniversaires ou événements rattachés à une journée entière

Cet exemple montre l'utilisation du paramètre "VALUE" pour lier un "VEVENT" à un jour plutôt qu'une heure spécifique.

```

BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:PUBLISH
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
DTSTAMP:19970614T190000Z
UID:0981234-1234234-23@exemple.com
DTSTART;VALUE=DATE:19970714
RRULE:FREQ=YEARLY;INTERVAL=1
SUMMARY: Bastille Day
END:VEVENT
END:VCALENDAR

```

4.2 Exemples d'événement de groupe

Les événements de groupe sont distingués des événements publiés en ce qu'ils ont des "Participants" et qu'il y a une interaction entre les "Participants" et l'"Organisateur" par rapport à l'événement. L'individu "A" demande une réunion entre les individus "A", "B", "C", et "D". L'individu "B" confirme sa présence à la réunion. L'individu "C" refuse de participer. L'individu "D" confirme une tentative de participation. Le tableau suivant illustre le flux de messages entre ces individus. "A", le CU qui programme la réunion, est appelé l'"Organisateur".

Action	"Organisateur"	Participants
Initie une demande de réunion	"A" envoie un message REQUEST à "B", "C", et "D".	
Accepte la demande de réunion		"B" envoie un message REPLY à "A" avec son paramètre ATTENDEE PARTSTAT réglé à ACCEPTED.
Refuse la demande de réunion		"C" envoie un message REPLY à "A" avec son paramètre ATTENDEE PARTSTAT réglé à DECLINED.
Tente d'accepter la demande de réunion		"D" envoie un message REPLY à "A" avec son paramètre ATTENDEE PARTSTAT réglé à TENTATIVE.
Confirme l'état de la réunion avec les participants	"A" envoie un message REQUEST à "B" et "D" avec des informations mises à jour.	

4.2.1 Demande d'événement de groupe

Un échantillon de demande de réunion est envoyé de "A" à "B", "C", et "D". Une copie de la demande est aussi envoyée à "E" mais il n'est pas supposé assister et n'a pas besoin de répondre. "E" illustre comment les CUA pourraient mettre en œuvre une caractéristique de type "pour information". Noter l'utilisation du paramètre "ROLE". La valeur par défaut pour le paramètre "ROLE" est "REQ-PARTICIPANT" et n'a pas besoin d'être mentionnée. Dans ce cas, on utilise la valeur "NON-PARTICIPANT" pour indiquer que "E" est un CU non assistant. Le paramètre n'est pas nécessaire sur les autres "Participants" car "PARTICIPANT" est la valeur par défaut.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED;CN=A:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL;CN=B:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL;CN=C:mailto:c@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL;CN=Hal:mailto:d@exemple.com
ATTENDEE;RSVP=FALSE;CUTYPE=ROOM:conf_big@exemple.com
ATTENDEE;ROLE=NON-PARTICIPANT;RSVP=FALSE:mailto:e@exemple.com
DTSTAMP:19970611T190000Z
DTSTART:19970701T200000Z
DTEND:19970701T210000Z
SUMMARY:Conference
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:0
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

4.2.2 Réponse à une demande d'événement de groupe

Le "participant" "B" accepte la réunion.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VEVENT
ATTENDEE;PARTSTAT=ACCEPTED:mailto:b@exemple.com
ORGANIZER:mailto:a@exemple.com
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:0
REQUEST-STATUS:2.0;Success
DTSTAMP:19970612T190000Z
END:VEVENT
END:VCALENDAR
```

"B" pourrait avoir refusé la réunion ou indiqué une tentative d'acceptation en réglant le paramètre "ATTENDEE" "PARTSTAT" à, respectivement, "DECLINED" ou "TENTATIVE". Aussi, "REQUEST-STATUS" n'est pas exigé dans les transactions réussies.

4.2.3 Mise à jour d'un événement

L'événement est déplacé à une heure différente. La combinaison de la propriété "UID" (inchangée) et de "SEQUENCE" (monté à 1) indique la mise à jour.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
```

```

BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:c@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL;CN=Hal:mailto:d@exemple.com
ATTENDEE;ROLE=NON-PARTICIPANT;RSVP=FALSE;
CUTYPE=ROOM:mailto:conf@exemple.com
ATTENDEE;ROLE=NON-PARTICIPANT;RSVP=FALSE:mailto:e@exemple.com
DTSTART:19970701T180000Z
DTEND:19970701T190000Z
SUMMARY:Phone Conference
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:1
DTSTAMP:19970613T190000Z
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR

```

4.2.4 Contre proposition d'événement

"A" envoie une "REQUEST" à "B" et "C". "B" fait une contre proposition à "A" pour changer l'heure et le lieu. "A" envoie la "REQUEST" suivante :

```

BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//FR
METHOD:REQUEST
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:c@exemple.com
DTSTART:19970701T190000Z
DTEND:19970701T200000Z
SUMMARY: Discussion des mérites des résultats de l'élection
LOCATION: Salle de conférence verte
UID:calsrv.exemple.com-873970198738777a@exemple.com
SEQUENCE:0
DTSTAMP:19970611T190000Z
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR

```

"B" envoie "COUNTER" à "A", demandant un changement d'heure et de lieu. "B" utilise la propriété "COMMENT" pour communiquer la raison du changement. Noter que la propriété "SEQUENCE" n'est pas incrémentée sur un "COUNTER".

```

BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//FR
METHOD:COUNTER
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:c@exemple.com
DTSTART:19970701T160000Z
DTEND:19970701T170000Z
DTSTAMP:19970612T190000Z
SUMMARY: Discussion des mérites du résultat de l'élection
LOCATION: Salle de conférence bleue

```

COMMENT: Cette heure me convient mieux et je pense que la grande salle de conférence est trop grande
 UID:calstrv.exemple.com-873970198738777a@exemple.com
 SEQUENCE:0
 END:VEVENT
 END:VCALENDAR

"A" accepte les changements de "B". Pour accepter une contre proposition, l'"Organisateur" envoie une nouvelle "REQUEST" d'événement avec un numéro de séquence incrémenté.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//FR
METHOD:REQUEST
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:c@exemple.com
DTSTAMP:19970613T190000Z
DTSTART:19970701T160000Z
DTEND:19970701T170000Z
SUMMARY: Discussion des mérites du résultat de l'élection – changée pour satisfaire le programme de B.
LOCATION: Salle de conférence bleue
UID:calstrv.exemple.com-873970198738777@exemple.com
SEQUENCE:1
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

Ici, "A" rejette la contre proposition de "B".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//FR
METHOD:DECLINECOUNTER
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:b@exemple.com
COMMENT: Désolé, je ne peux pas changer l'heure de cette réunion.
UID:calstrv.exemple.com-873970198738777@exemple.com
SEQUENCE:0
DTSTAMP:19970614T190000Z
END:VEVENT
END:VCALENDAR
```

4.2.5 Délégation d'un événement

Quand il délègue une demande d'événement à un autre "Utilisateur de calendrier", le "Déléguant" doit à a fois mettre à jour l'"Organisateur" avec une "REPLY" et envoyer une demande au "Délégué". Il n'y a actuellement pas de limitation de la part du protocole à une profondeur de délégation. Il est possible au délégué original de déléguer la réunion à quelqu'un d'autre, et ainsi de suite. Quand une demande est déléguée d'un CUA à un autre, il y a un certain nombre de responsabilités exigées du "Déléguant". Le "Déléguant" DOIT :

- o Envoyer une "REPLY" à l'"Organisateur" avec les mises à jour suivantes :
 - A. Le paramètre "PARTSTAT" de la propriété "ATTENDEE" du "Déléguant" est réglé à "DELEGATED" et le paramètre "DELEGATED-TO" est réglé à l'adresse du "Délégué".
 - B. Ajouter une propriété "ATTENDEE" supplémentaire pour le "Délégué" avec le paramètre de propriété "DELEGATED-FROM" réglé au "Déléguant".
 - C. Indiquer si ils veulent continuer de recevoir les mises à jour quand l'"Organisateur" envoie les versions mises à jour de l'événement. Régler le paramètre de propriété "RSVP" à "TRUE" cause l'envoi des mises à jour ; le régler à "FALSE" cause la fin de l'envoi des mises à jour. Noter que dans les deux cas, si le "Délégué" refuse l'invitation, cela sera notifié au "Déléguant".

- o Le "Délégant" DOIT aussi envoyer une copie de la méthode "REQUEST" originale au "Délégué", avec les changements (A) et (B), appliqués comme précisé ci-dessus.

Si le "Délégué" refuse la réunion, l'"Organisateur" DOIT envoyer une mise à jour de "REQUEST" au "Délégant" afin que le "Délégant" puisse choisir de déléguer la "REQUEST" à un autre CUA.

Action	"Organisateur"	"Participants"
Initie une demande de réunion Déléguer : "C" délègue à "E"	"A" envoie un message REQUEST à "B" et "C".	"C" envoie une REPLY à "A" avec le paramètre ATTENDEE PARTSTAT réglé à DELEGATED et avec une nouvelle propriété ATTENDEE pour "E". Le paramètre ATTENDEE DELEGATED-FROM de "E" est réglé à "C". Le paramètre ATTENDEE DELEGATED-TO de "C" est réglé à "E". "C" envoie le message REQUEST à "E" avec les informations originales de demande de réunion. Le paramètre de propriété PARTSTAT pour "C" est réglé à DELEGATED et le paramètre DELEGATED-TO est réglé à l'adresse de "E". Une propriété ATTENDEE est ajoutée pour "E" et le paramètre DELEGATED-FROM est réglé à l'adresse de "C".
Confirmer la participation à la réunion Facultatif : redistribuer la réunion aux participants	"A" envoie le message REQUEST à "B", "C", et "E".	"E" envoie un message REPLY à "A", et facultativement à "C", avec son paramètre de propriété PARTSTAT réglé à ACCEPTED.

"C" répond à l'"Organisateur".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@example.com
ATTENDEE;PARTSTAT=DELEGATED;DELEGATED-TO="mailto:e@example.com":mailto:c@example.com
UID:calsrv.example.com-873970198738777@example.com
SEQUENCE:0
REQUEST-STATUS:2.0;Success
DTSTAMP:19970611T190000Z
END:VEVENT
END:VCALENDAR
```

Le "participant" "C" délègue sa présence à la réunion à "E".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@example.com
ATTENDEE;PARTSTAT=DELEGATED;DELEGATED-TO="mailto:e@example.com":mailto:c@example.com
ATTENDEE;RSVP=TRUE;
DELEGATED-FROM="mailto:c@example.com":mailto:e@example.com
DTSTART:19970701T180000Z
DTEND:19970701T200000Z
SUMMARY:Phone Conference
UID:calsrv.example.com-873970198738777@example.com
SEQUENCE:0
```

```
STATUS:CONFIRMED
DTSTAMP:19970611T190000Z
END:VEVENT
END:VCALENDAR
```

4.2.6 Le délégué accepte la réunion

Pour accepter une réunion déléguée, le délégué, "E", envoie le message suivant à "A" et "C".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;PARTSTAT=ACCEPTED;DELEGATED-
FROM="mailto:c@exemple.com":mailto:e@exemple.com
ATTENDEE;PARTSTAT=DELEGATED;
DELEGATED-TO="mailto:e@exemple.com":mailto:c@exemple.com
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:0
REQUEST-STATUS:2.0;Success
DTSTAMP:19970614T190000Z
END:VEVENT
END:VCALENDAR
```

4.2.7 Le délégué refuse la réunion

Dans cet exemple, le "Délégué" refuse la demande de réunion et règle le paramètre "PARTSTAT" de la propriété "ATTENDEE" à "DECLINED". L'"Organisateur" DEVRAIT renvoyer la "REQUEST" à "C" avec le paramètre "PARTSTAT" du "Délégué" réglé à "DECLINED". Cela fait savoir au "Déléguant" que le "Délégué" a refusé et donne une opportunité au "Déléguant" d'accepter la demande ou de la déléguer à un autre CU.

"E" répond à "A" et "C". Noter l'utilisation de la propriété "COMMENT" avec laquelle "E" indique pourquoi la délégation a été refusée.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//FR
METHOD:REPLY
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;PARTSTAT=DELEGATED;
DELEGATED-TO="mailto:e@exemple.com":mailto:c@exemple.com
ATTENDEE;PARTSTAT=DECLINED;
DELEGATED-FROM="mailto:c@exemple.com":mailto:e@exemple.com
COMMENT: Désolé, je ne serai pas en ville à cette heure là.
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:0
REQUEST-STATUS:2.0;Success
DTSTAMP:19970614T190000Z
END:VEVENT
END:VCALENDAR
```

"A" renvoie la méthode "REQUEST" à "C". "A" peut aussi souhaiter exprimer le fait que l'élément a été délégué dans la propriété "COMMENT".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
```

```

BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;PARTSTAT=DECLINED;
DELEGATED-FROM="mailto:c@exemple.com":mailto:e@exemple.com
ATTENDEE;RSVP=TRUE:mailto:c@exemple.com
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:0
SUMMARY:Phone Conference
DTSTART:19970701T180000Z
DTEND:19970701T200000Z
DTSTAMP:19970614T200000Z
COMMENT:DELEGATE (ATTENDEE mailto:e@exemple.com) DECLINED YOUR
INVITATION
END:VEVENT
END:VCALENDAR

```

4.2.8 Transmission d'une demande d'événement

Le protocole n'empêche pas un "participant" de "transmettre" un composant de calendrier "VEVENT" à d'autres "utilisateurs de calendrier". La transmission diffère de la délégation en ce que l'"Utilisateur de calendrier" qui reçoit la transmission (qu'on appelle souvent un "trouble-fête") ne remplace pas l'"Utilisateur de calendrier" qui transmet. Les mises en œuvre ne sont pas obligées d'ajouter le "trouble-fête" à la liste des "participants", et donc il n'est pas garanti qu'un "trouble-fête" recevra des mises à jour supplémentaires de l'événement. L'"Utilisateur de calendrier" qui transmet NE DEVRAIT PAS ajouter le "trouble-fête" à la liste des "Participants". L'"Organisateur" PEUT ajouter l'"Utilisateur de calendrier" qui reçoit la transmission à la liste des "Participants".

4.2.9 Annulation d'un événement de groupe

L'individu "A" demande une réunion entre les individus "A", "B", "C", et "D". L'individu "B" décline l'invitation à la réunion. L'individu "A" décide d'annuler la réunion. Le tableau qui suit illustre la séquence des messages qui vont être échangés entre ces individus.

Les messages relatifs à un événement précédemment annulé (la valeur de propriété "SEQUENCE" est inférieure à la valeur de propriété "SEQUENCE" du message "CANCEL") DOIVENT être ignorés.

Action	Organisateur	Participant
Initie une demande de réunion	"A" envoie un message REQUEST à "B", "C", et "D".	
Refuse la demande de réunion		"B" envoie un message REPLY à "A" avec le paramètre PARTSTAT réglé à DECLINED.
Annule la réunion	"A" envoie un message CANCEL à "B", "C", et "D".	

Cet exemple montre comment "A" annule l'événement.

```

BEGIN:VCALENDAR
PROID:-//Example/ExampleCalendarClient//FR
METHOD:CANCEL
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;CUTYPE=INDIVIDUAL;mailto:a@exemple.com
ATTENDEE;CUTYPE=INDIVIDUAL;mailto:b@exemple.com
ATTENDEE;CUTYPE=INDIVIDUAL;mailto:c@exemple.com
ATTENDEE;CUTYPE=INDIVIDUAL;mailto:d@exemple.com
COMMENT: M. B ne peut pas venir. Il pleut. Annulons.
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:1
STATUS:CANCELLED
DTSTAMP:19970613T190000Z
END:VEVENT
END:VCALENDAR

```

4.2.10 Suppression de participants

"A" veut retirer "B" d'une réunion. Il fait cela en envoyant un "CANCEL" à "B" et en retirant "B" de la liste des "Participants" dans la copie maîtresse de l'événement.

Action	Organisateur	Participant
Retire "B" des participants	"A" envoie un message CANCEL à "B".	
Met à jour la copie maîtresse de l'événement	"A" envoie facultativement un événement mis à jour aux participants restants.	

La réunion inclut à l'origine "A", "B", "C", et "D". L'exemple ci-dessous montre le "CANCEL" que "A" envoie à "B". Noter que dans l'exemple ci-dessous, la propriété "STATUS" est omise. Ceci est utilisé quand la réunion elle-même est annulée et non quand l'intention est de retirer un "Participant" de l'événement.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//FR
METHOD:CANCEL
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
COMMENT:Tu es décroché pour cette réunion
UID:calsrv.exemple.com-873970198738777@exemple.com
DTSTAMP:19970613T193000Z
SEQUENCE:1
END:VEVENT
END:VCALENDAR
```

La copie maîtresse mise à jour de l'événement est montrée ci-dessous. L'"Organisateur" PEUT renvoyer l'événement mis à jour aux "Participants" restants. Noter que "B" a été retiré.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;CUTYPE=INDIVIDUAL:mailto:c@exemple.com
ATTENDEE;CUTYPE=INDIVIDUAL:mailto:d@exemple.com
ATTENDEE;CUTYPE=ROOM:mailto:cr_big@exemple.com
ATTENDEE;ROLE=NON-PARTICIPANT;
RSVP=FALSE:mailto:e@exemple.com
DTSTAMP:19970611T190000Z
DTSTART:19970701T200000Z
DTEND:19970701T203000Z
SUMMARY:Phone Conference
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:2
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

4.2.11 Remplacement de l'organisateur

Le scénario de cet exemple commence avec "A" comme "organisateur" pour une réunion récurrente avec "B", "C", et "D". "A" reçoit une offre de nouveau travail dans un autre pays et se trouve hors course. "A" ne laisse pas d'adresse de transmission ou de moyen d'être joint. En utilisant une communication hors bande, les autres "participants" apprennent finalement ce qui s'est passé et trouvent un accord pour que "B" devienne le nouvel "Organisateur" de la réunion. Pour faire cela, "B" envoie une nouvelle version de l'événement et les autres "Participants" s'accordent pour accepter "B" comme

nouvel "Organisateur". "B" retire aussi "A" de l'événement.

Quand l'"organisateur" est remplacé, la valeur de propriété "SEQUENCE" DOIT être incrémentée. Voici le message que "B" envoie à "C" et "D".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:b@exemple.com
ATTENDEE;ROLE=CHAIR;STATUS=ACCEPTED:mailto:b@exemple.com
ATTENDEE;CUTYPE=INDIVIDUAL:mailto:c@exemple.com
ATTENDEE;CUTYPE=INDIVIDUAL:mailto:d@exemple.com
DTSTAMP:19970611T190000Z
DTSTART:19970701T200000Z
DTEND:19970701T203000Z
RRULE:FREQ=WEEKLY
SUMMARY:Phone Conference
UID:123456@exemple.com
SEQUENCE:1
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

4.3 Exemples de temps occupés

Les objets de temps occupés peuvent être utilisés de plusieurs façons. D'abord, un CU peut demander les temps d'occupation d'un autre CU pour une gamme de temps spécifique. Il peut être répondu à cette demande avec une réponse de temps d'occupation "REPLY". De plus, un CU peut simplement publier ses temps d'occupation pour un certain intervalle de temps et pointer les autres CU sur le lieu de publication. Les exemples qui suivent illustrent les deux scénarios.

4.3.1 Publication des temps occupés

L'individu "A" publie ses temps d'occupation pour une semaine.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
METHOD:PUBLISH
BEGIN:VFREEBUSY
DTSTAMP:19980101T124100Z
ORGANIZER:mailto:a@exemple.com
DTSTART:19980101T124200Z
DTEND:19980108T124200Z
FREEBUSY:19980101T180000Z/19980101T190000Z
FREEBUSY:19980103T020000Z/19980103T050000Z
FREEBUSY:19980107T020000Z/19980107T050000Z
FREEBUSY:19980113T000000Z/19980113T010000Z
FREEBUSY:19980115T190000Z/19980115T200000Z
FREEBUSY:19980115T220000Z/19980115T230000Z
FREEBUSY:19980116T013000Z/19980116T043000Z
END:VFREEBUSY
END:VCALENDAR
```

4.3.2 Demande des temps occupés

L'individu "A" demande les temps d'occupation des individus "B" et "C". Les individus "B" et "C" répondent avec leurs données de temps d'occupation à "A". Le tableau suivant illustre la séquence des messages qui seraient échangés entre ces individus.

Action	Organisateur	Participant
Initie une demande de temps d'occupation	"A" envoie un message REQUEST à "B" et "C".	
Réponse à la demande BUSY par des données de temps BUSY		"B" envoie un message REPLY à "A" avec ses données de temps d'occupation.

"A" envoie une "REQUEST" à "B" et "C" pour leurs temps d'occupation.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VFREEBUSY
ORGANIZER:mailto:a@example.com
ATTENDEE;ROLE=CHAIR:mailto:a@example.com
ATTENDEE:mailto:b@example.com
ATTENDEE:mailto:c@example.com
DTSTAMP:19970613T190000Z
DTSTART:19970701T080000Z
DTEND:19970701T200000
UID:calsrv.example.com-873970198738777@example.com
END:VFREEBUSY
END:VCALENDAR
```

4.3.3 Réponse à une demande des temps occupés

"B" envoie un type de méthode "REPLY" d'un composant de calendrier "VFREEBUSY" à "A".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VFREEBUSY
ORGANIZER:mailto:a@example.com
ATTENDEE:mailto:b@example.com
DTSTART:19970701T080000Z
DTEND:19970701T200000Z
UID:calsrv.example.com-873970198738777@example.com
FREEBUSY:19970701T090000Z/PT1H,19970701T140000Z/PT30M
DTSTAMP:19970613T190030Z
END:VFREEBUSY
END:VCALENDAR
```

"B" est occupé de 09:00 à 10:00 et de 14:00 à 14:30.

4.4 Exemples d'événement récurrent et de zone horaire

4.4.1 Événement récurrent s'étendant sur plusieurs zones horaires

Cet événement décrit une conférence téléphonique hebdomadaire. Les "participants" sont chacun dans une zone horaire différente.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VTIMEZONE
TZID:America-SanJose
TZURL:http://exemple.com/tz/America-SanJose
```

```

BEGIN:STANDARD
DTSTART:19671029T020000
RRULE:FREQ=YEARLY;BYDAY=-1SU;BYMONTH=10
TZOFFSETFROM:-0700
TZOFFSETTO:-0800
TZNAME:PST
END:STANDARD
BEGIN:DAYLIGHT
DTSTART:19870405T020000
RRULE:FREQ=YEARLY;BYDAY=1SU;BYMONTH=4
TZOFFSETFROM:-0800
TZOFFSETTO:-0700
TZNAME:PDT
END:DAYLIGHT
END:VTIMEZONE
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED;
CUTYPE=INDIVIDUAL:a@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:b@exemple.fr
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:c@exemple.jp
DTSTAMP:19970613T190030Z
DTSTART;TZID=America-SanJose:19970701T140000
DTEND;TZID=America-SanJose:19970701T150000
RRULE:FREQ=WEEKLY;COUNT=20;WKST=SU;BYDAY=TU
RDATE;TZID=America-SanJose:19970910T140000
EXDATE;TZID=America-SanJose:19970909T140000
EXDATE;TZID=America-SanJose:19971028T140000
SUMMARY:Weekly Phone Conference
UID:calsrv.exemple.com-873970198738777@exemple.com
SEQUENCE:0
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR

```

Le premier composant de cet objet iCalendar est le composant de zone horaire. La date "DTSTART" coïncide avec la première instance de la propriété "RRULE".

La réunion récurrente est définie dans une zone horaire particulière, probablement celle de l'organisateur. Le client pour chaque "Participant" a la responsabilité de déterminer l'heure de récurrence dans la zone horaire du "Participant".

L'événement répétitif commence le jeudi 1er juillet 1997 à 2:00 pm PDT (UTC-7). Le "Participant" B@exemple.fr est en France, où l'heure locale à cette date est 9 heures en avance de PDT, soit 23:00 CEST (UTC+2). Le "Participant" C@exemple.jp est au Japon, où l'heure locale est en retard de 16 heures sur PDT, soit mercredi 2 juillet à 06:00 JST (UTC+9). L'événement se répète chaque semaine les jeudis (dans PST/PDT). La propriété "RRULE" résulte en 20 instances. La dernière instance tombe le jeudi 11 novembre 1997 2:00pm PST. La propriété "RDATE" ajoute une autre instance : WED, 10-SEP-1997 2:00 PM PDT.

Il y a aussi deux dates d'exception à la règle de récurrence. La première est :

- o TUE, 09-SEP-1997 14:00 PDT (UTC-7)
- o TUE, 09-SEP-1997 23:00 CEST (UTC+2)
- o WED, 10-SEP-1997 06:00 JST (UTC+9)

et la seconde est :

- o TUE, 28-OCT-1997 14:00 PST (UTC-8)
- o TUE, 28-OCT-1997 23:00 CET (UTC+1)
- o WED, 29-OCT-1997 07:00 JST (UTC+9)

4.4.2 Modification d'une instance récurrente

Dans cet exemple, l'"Organisateur" lance une réunion récurrente. Ensuite, l'"Organisateur" change une instance de

l'événement en changeant la propriété "DTSTART". Noter l'utilisation de la propriété "RECURRENCE-ID" et de la propriété "SEQUENCE" dans la seconde demande.

Demande originale :

```
BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:guid-1@exemple.com
SEQUENCE:0
RRULE:FREQ=MONTHLY;BYMONTHDAY=1;UNTIL=19980901T210000Z
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
ATTENDEE:mailto:c@exemple.com
ATTENDEE:mailto:d@exemple.com
DESCRIPTION:IETF-C&S Conference Call
CLASS:PUBLIC
SUMMARY:IETF Calendaring Working Group Meeting
DTSTART:19970601T210000Z
DTEND:19970601T220000Z
LOCATION:Conference Call
DTSTAMP:19970526T083000Z
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

La demande d'événement ci-dessous est le changement de l'heure d'une instance spécifique. Cela déplace l'instance du 1er juillet au 3 juillet.

```
BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:guid-1@exemple.com
RECURRENCE-ID:19970701T210000Z
SEQUENCE:1
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
ATTENDEE:mailto:c@exemple.com
ATTENDEE:mailto:d@exemple.com
DESCRIPTION:IETF-C&S Conference Call
CLASS:PUBLIC
SUMMARY:IETF Calendaring Working Group Meeting
DTSTART:19970703T210000Z
DTEND:19970703T220000Z
LOCATION:Conference Call
DTSTAMP:19970626T093000Z
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

4.4.3 Annulation d'une instance

Dans cet exemple, l'"Organisateur" d'un événement récurrent supprime l'instance du 1er août.

```
BEGIN:VCALENDAR
```

```

METHOD:CANCEL
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:guid-1@exemple.com
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
ATTENDEE:mailto:c@exemple.com
ATTENDEE:mailto:d@exemple.com
RECURRENCE-ID:19970801T210000Z
SEQUENCE:2
STATUS:CANCELLED
DTSTAMP:19970721T093000Z
END:VEVENT
END:VCALENDAR

```

4.4.4 Annulation d'un événement récurrent

Dans cet exemple, l'"Organisateur" souhaite annuler l'événement récurrent entier et toutes les exceptions.

```

BEGIN:VCALENDAR
METHOD:CANCEL
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:guid-1@exemple.com
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
ATTENDEE:mailto:c@exemple.com
ATTENDEE:mailto:d@exemple.com
DTSTAMP:19970721T103000Z
STATUS:CANCELLED
SEQUENCE:3
END:VEVENT
END:VCALENDAR

```

4.4.5 Changer toutes les instances futures

Cet exemple change le lieu de la réunion prévue dans un appel à conférence à Seattle, commençant le 1er septembre et s'étendant à toutes les instances futures.

```

BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:guid-1@exemple.com
RECURRENCE-ID;THISANDFUTURE:19970901T210000Z
SEQUENCE:3
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE:mailto:c@exemple.com
ATTENDEE;RSVP=TRUE:mailto:d@exemple.com
DESCRIPTION:IETF-C&S Discussion
CLASS:PUBLIC
SUMMARY:IETF Calendaring Working Group Meeting
DTSTART:19970901T210000Z
DTEND:19970901T220000Z

```

LOCATION:Building 32, Microsoft, Seattle, WA
 DTSTAMP:19970526T083000Z
 STATUS:CONFIRMED
 END:VEVENT
 END:VCALENDAR

4.4.6 Ajouter une nouvelle instance à un événement récurrent

Cet exemple ajoute une instance supplémentaire exceptionnelle à l'événement récurrent. L'"Organisateur" ajoute une seconde réunion de juillet, le 15.

```
BEGIN:VCALENDAR
METHOD:ADD
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:123456789@exemple.com
SEQUENCE:4
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE:mailto:c@exemple.com
ATTENDEE;RSVP=TRUE:mailto:d@exemple.com
DESCRIPTION:IETF-C&S Conference Call
CLASS:PUBLIC
SUMMARY:IETF Calendaring Working Group Meeting
DTSTART:19970715T210000Z
DTEND:19970715T220000Z
LOCATION:Conference Hall
DTSTAMP:19970629T093000Z
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

4.4.7 Ajouter une nouvelle série d'instances à un événement récurrent

Le scénario de cet exemple implique une réunion à venir, prévue à l'origine pour se produire chaque mardi. L'"Organisateur" décide ensuite que les réunions doivent être les mardis et jeudis.

L'événement original :

```
BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:123456789@exemple.com
SEQUENCE:0
RRULE:WKST=SU;BYDAY=TU;FREQ=WEEKLY
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
SUMMARY:Review Accounts
DTSTART:19980303T210000Z
DTEND:19980303T220000Z
LOCATION:The White Room
DTSTAMP:19980301T093000Z
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

L'événement entier peut être reprogrammé en utilisant une "REQUEST". C'est fait en utilisant le "UID" de l'événement pour reprogrammer et inclure la "RRULE" modifiée. Noter que comme c'est une reprogrammation entière de l'événement, toute information spécifique d'une instance va être perdue, sauf explicitement incluse avec la "REQUEST" de mise à jour.

```
BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:123456789@exemple.com
SEQUENCE:7
RRULE:WKST=SU;BYDAY=TU,TH;FREQ=WEEKLY
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
SUMMARY:Review Accounts
DTSTART:19980303T210000Z
DTEND:19980303T220000Z
DTSTAMP:19980303T193000Z
LOCATION:The White Room
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

4.4.8 Rafraîchissement d'un événement récurrent

La série d'exemples suivante illustre comment un "Organisateur" répondrait à une "REFRESH" soumise par un "Participant" après une série de modifications spécifiques de l'instance. Pour porter tous les changements spécifiques d'instance, l'"Organisateur" doit fournir la dernière description d'événement et des instances pertinentes. Les trois premiers exemples montrent l'historique, incluant la demande "VEVENT" initiale et les changements d'instances suivantes, et finalement la "REFRESH".

Demande d'origine :

```
BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:123456789@exemple.com
SEQUENCE:0
RDATE:19980304T180000Z
RDATE:19980311T180000Z
RDATE:19980318T180000Z
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
SUMMARY:Review Accounts
DTSTART:19980304T180000Z
DTEND:19980304T200000Z
DTSTAMP:19980303T193000Z
LOCATION:Conference Room A
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

L'organisateur change le lieu et l'heure de la deuxième instance :

```
BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
```

```

VERSION:2.0
BEGIN:VEVENT
UID:123456789@exemple.com
SEQUENCE:1
RECURRENCE-ID:19980311T180000Z
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
SUMMARY:Review Accounts
DTSTART:19980311T160000Z
DTEND:19980311T180000Z
DTSTAMP:19980306T193000Z
LOCATION:The Small conference room
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR

```

L'organisateur ajoute une quatrième instance de la réunion en utilisant la méthode "ADD".

```

BEGIN:VCALENDAR
METHOD:ADD
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:123456789@exemple.com
SEQUENCE:2
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
SUMMARY:Review Accounts
DTSTART:19980315T180000Z
DTEND:19980315T200000Z
DTSTAMP:19980307T193000Z
LOCATION:Conference Room A
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR

```

Si "B" demande une "REFRESH", "A" répond par ce qui suit pour capturer toutes les données spécifiques d'instance. Dans ce cas, sont inclus la demande initiale et un "VEVENT" supplémentaire qui spécifie les données spécifiques d'instance. Parce qu'ils sont tous deux du même type (ce sont tous deux des "VEVENT") ils peuvent être envoyés dans le même objet iCalendar.

```

BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:123456789@exemple.com
SEQUENCE:2
RDATE:19980304T180000Z
RDATE:19980311T160000Z
RDATE:19980315T180000Z
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
SUMMARY:Review Accounts
DTSTART:19980304T180000Z
DTEND:19980304T200000Z
DTSTAMP:19980303T193000Z
LOCATION:Conference Room A

```

```

STATUS:CONFIRMED
END:VEVENT
BEGIN:VEVENT
SEQUENCE:2
UID:123456789@exemple.com
RECURRENCE-ID:19980311T160000Z
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
SUMMARY:Review Accounts
DTSTART:19980311T160000Z
DTEND:19980304T180000Z
DTSTAMP:19980306T193000Z
LOCATION:The Small conference room
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR

```

4.4.9 Contre proposition sur une instance d'un événement récurrent

Dans cet exemple, un des "Participants" conteste la propriété "DTSTART" de la seconde réunion de juillet proposée.

```

BEGIN:VCALENDAR
METHOD:COUNTER
PRODID:-//Example/ExampleCalendarClient//FR
VERSION:2.0
BEGIN:VEVENT
UID:guid-1@exemple.com
RECURRENCE-ID:19970715T210000Z
SEQUENCE:4
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;RSVP=TRUE:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE:mailto:c@exemple.com
ATTENDEE;RSVP=TRUE:mailto:d@exemple.com
DESCRIPTION:Conférence téléphonique de IETF-C&S
CLASS:PUBLIC
SUMMARY:Réunion du groupe de travail Calendrier de l'IETF
DTSTART:19970715T220000Z
DTEND:19970715T230000Z
LOCATION:Conférence téléphonique
COMMENT:Pouvons nous retarder cela d'une heure ? J'ai un conflit
DTSTAMP:19970629T094000Z
END:VEVENT
END:VCALENDAR

```

4.4.10 Réponse d'erreur à une demande

L'exemple suivant illustre un scénario où une réunion est proposée qui contient une propriété non prise en charge et une mauvaise propriété.

Demande originale :

```

BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:guid-1@exemple.com
SEQUENCE:0
RRULE:FREQ=MONTHLY;BYMONTHDAY=1
ORGANIZER:mailto:a@exemple.com

```

```

ATTENDEE;ROLE=CHAIR:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE:mailto:c@exemple.com
ATTENDEE;RSVP=TRUE:mailto:d@exemple.com
DESCRIPTION:IETF-C&S Conference Call
CLASS:PUBLIC
SUMMARY:IETF Calendaring Working Group Meeting
DTSTART:19970601T210000Z
DTEND:19970601T220000Z
DTSTAMP:19970602T094000Z
LOCATION:Conference Call
STATUS:CONFIRMED
FOO:BAR
END:VEVENT
END:VCALENDAR

```

"B" répond pour indiquer que "RRULE" n'est pas acceptée et qu'il a rencontré une propriété non reconnue.

```

BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
REQUEST-STATUS:3.0;Invalid Property Name;FOO
UID:guid-1@exemple.com
SEQUENCE:0
DTSTAMP:19970603T094000Z
END:VEVENT
END:VCALENDAR

```

4.5 Exemples de choses à faire de groupe

L'individu "A" crée une tâche de groupe à laquelle les individus "A", "B", "C", et "D" vont participer. L'individu "B" confirme l'acceptation de la tâche. L'individu "C" refuse la tâche. L'individu "D" accepte provisoirement la tâche. Le tableau suivant illustre la séquence de messages qui seraient échangés entre ces individus. L'individu "A" produit ensuite une méthode "REQUEST" pour obtenir l'état de la chose à faire de chaque participant. La réponse indique l'état d'achèvement des "participants" individuels. Le tableau ci-dessous illustre le flux des messages.

Action	Organisateur	Participant
Initie une demande de chose à faire	"A" envoie a REQUEST message to "B", "C", et "D".	
Accepte la demande de chose à faire		"B" envoie un message REPLY à "A" avec son paramètre PARTSTAT réglé à ACCEPTED.
Refuse la demande de chose à faire		"C" envoie un message REPLY à "A" avec son paramètre PARTSTAT réglé à DECLINED.
Accepte provisoirement la demande de chose à faire		"D" envoie un message REPLY à "A" avec son paramètre PARTSTAT réglé à TENTATIVE.
Vérifie l'état d'achèvement du participant	"A" envoie un message REQUEST à "B" et "D" avec les informations courantes.	
Le participant indique le pourcentage achevé		"B" envoie un message REPLY indiquant le pourcentage achevé.
Le participant indique l'achèvement		"D" envoie un message REPLY indiquant l'achèvement.

4.5.1 Demande VTODO

Échantillon de "REQUEST" pour un composant de calendrier "VTODO" que "A" envoie à "B", "C", et "D".

```

BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VTODO
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE:mailto:c@exemple.com
ATTENDEE;RSVP=TRUE:mailto:d@exemple.com
DTSTART:19970701T170000Z
DUE:19970722T170000Z
PRIORITY:1
SUMMARY:Create the requirements document
UID:calsrv.exemple.com-873970198738777-00@exemple.com
SEQUENCE:0
DTSTAMP:19970717T200000Z
STATUS:NEEDS-ACTION
END:VTODO
END:VCALENDAR

```

4.5.2 Réponse VTODO

"B" accepte la chose à faire.

```

BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VTODO
ORGANIZER:mailto:a@exemple.com
ATTENDEE;PARTSTAT=ACCEPTED:mailto:b@exemple.com
UID:calsrv.exemple.com-873970198738777-00@exemple.com
COMMENT:I'll send you my input by email
SEQUENCE:0
DTSTAMP:19970717T203000Z
REQUEST-STATUS:2.0;Success
END:VTODO
END:VCALENDAR

```

"B" pourrait avoir refusé le "VTODO" ou indiqué une tentative d'acceptation en réglant la séquence de paramètre de propriété "PARTSTAT" à, respectivement, "DECLINED" ou "TENTATIVE".

4.5.3 Demande VTODO de mise à jour d'état

"A" demande l'état de tous les "participants".

```

BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VTODO
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR:mailto:a@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:b@exemple.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:d@exemple.com
UID:calsrv.exemple.com-873970198738777-00@exemple.com
SUMMARY:Create the requirements document
PRIORITY:1
SEQUENCE:0
STATUS:IN-PROCESS

```

```
DTSTART:19970701T170000Z
DTSTAMP:19970717T230000Z
END:VTODO
END:VCALENDAR
```

4.5.4 Réponse Pourcentage achevé

Une réponse indiquant que la tâche est en cours et que "B" a achevé à 75 % la part qui lui est allouée.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VTODO
ORGANIZER:mailto:a@exemple.com
ATTENDEE;PARTSTAT=IN-PROCESS:mailto:b@exemple.com
PERCENT-COMPLETE:75
UID:calstrv.exemple.com-873970198738777-00@exemple.com
DTSTAMP:19970717T233000Z
SEQUENCE:0
END:VTODO
END:VCALENDAR
```

4.5.5 Réponse : Achevé

Une réponse indique que "D" a achevé sa part de l'allocation.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VTODO
ORGANIZER:mailto:a@exemple.com
ATTENDEE;PARTSTAT=COMPLETED:mailto:d@exemple.com
UID:calstrv.exemple.com-873970198738777-00@exemple.com
DTSTAMP:19970717T233000Z
SEQUENCE:0
END:VTODO
END:VCALENDAR
```

4.5.6 Réponse VTODO mise à jour

L'organisateur "A" renvoie le composant de calendrier "VTODO". "A" règle l'achèvement complet de la chose à faire à 40 %.

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VTODO
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE;PARTSTAT=ACCEPTED;CUTYPE=INDIVIDUAL:mailto:b@exemple.com
ATTENDEE;PARTSTAT=COMPLETED;CUTYPE=INDIVIDUAL:mailto:d@exemple.com
DTSTART:19970701T170000Z
DUE:19970722T170000Z
PRIORITY:1
SUMMARY:Create the requirements document
UID:calstrv.exemple.com-873970198738777-00@exemple.com
SEQUENCE:1
DTSTAMP:19970718T100000Z
```

```
STATUS:IN-PROCESS
PERCENT-COMPLETE:40
END:VTODO
END:VCALENDAR
```

4.5.7 VTODD récurrents

Les exemples suivants se rapportent à des composants de calendrier "VTODO" récurrents.

4.5.7.1 Demande d'un VTODO récurrent

Dans cet exemple, "A" envoie un composant de calendrier "VTODO" récurrent à "B" et "D".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VTODO
ORGANIZER:mailto:a@example.com
ATTENDEE;ROLE=CHAIR:mailto:a@example.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:b@example.com
ATTENDEE;RSVP=TRUE;CUTYPE=INDIVIDUAL:mailto:d@example.com
RRULE:FREQ=MONTHLY;COUNT=10;BYDAY=1FR
DTSTART:19980101T100000Z
DUE:19980103T100000Z
SUMMARY:Send Status Reports to Area Managers
UID:cal.srv.example.com-873970198738777-00@example.com
SEQUENCE:0
DTSTAMP:19970717T200000Z
STATUS:NEEDS-ACTION
PRIORITY:1
END:VTODO
END:VCALENDAR
```

4.5.7.2 Replying to an Instance of a Recurring VTODO

Dans cet exemple, "B" met à jour "A" sur une seule instance du composant de calendrier "VTODO".

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REPLY
VERSION:2.0
BEGIN:VTODO
ATTENDEE;PARTSTAT=IN-PROCESS:mailto:b@example.com
PERCENT-COMPLETE:75
UID:cal.srv.example.com-873970198738777-00@example.com
DTSTAMP:19970717T233000Z
RECURRENCE-ID:19980101T170000Z
SEQUENCE:1
END:VTODO
END:VCALENDAR
```

4.6 Exemples de journal

L'objet iCalendar ci-dessous décrit une seule entrée de journal pour le 2 octobre 1997. La propriété "RELATED-TO" fait référence à l'événement de conférence téléphonique pour lequel le compte rendu a été fait.

```
BEGIN:VCALENDAR
METHOD:PUBLISH
PRODID:-//Example/ExampleCalendarClient//FR
VERSION:2.0
```

```

BEGIN:VJOURNAL
DTSTART:19971002T200000Z
DTSTAMP:19970717T233100Z
ORGANIZER:mailto:a@exemple.com
SUMMARY:Compte-rendu de la conférence téléphonique
DESCRIPTION:La réunion des éditeurs s'est tenue le 1er octobre 1997.
Les détails sont dans le document joint.
UID:0981234-1234234-2410@exemple.com
RELATED-TO:0981234-1234234-2402-35@exemple.com
ATTACH:ftp://ftp.exemple.com/pub/ed/minutes100197.txt
END:VJOURNAL
END:VCALENDAR

```

4.7 Autres exemples

4.7.1 Rafraîchissement d'événement

Rafraîchissement de l'événement qui a une valeur de propriété "UID" de "guid-1-12345@exemple.com":

```

BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REFRESH
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
ATTENDEE:mailto:c@exemple.com
ATTENDEE:mailto:d@exemple.com
UID:guid-1-12345@exemple.com
DTSTAMP:19970603T094000
END:VEVENT
END:VCALENDAR

```

4.7.2 Mauvais RECURRENCE-ID

Les instances de composants sont identifiées par la combinaison de "UID", "RECURRENCE-ID", et "SEQUENCE". Quand un "Organisateur" envoie un message iTIP à un "Participant", il y a trois cas où une instance ne peut pas être trouvée. Ce sont :

1. Le composant avec le "UID" et le "RECURRENCE-ID" référencés ont été trouvés mais le numéro de séquence dans le magasin de calendrier ne correspond pas à celui du message iTIP.
2. Le composant avec le "UID" référencé a été trouvé, les numéros de séquence correspondent, mais le "RECURRENCE-ID" ne peut pas être trouvé.
3. Le "UID" et les numéros de "SEQUENCE" sont trouvés mais le CUA ne prend pas en charge les récurrences.

Dans le cas (1), deux choses peuvent se produire. Si le numéro de "SEQUENCE" de l'instance de "Participant" est supérieur à celui du message de l'"Organisateur", le "Participant" a reçu un message hors séquence et DOIT l'ignorer. Si le numéro de "SEQUENCE" de l'instance de "Participant" est plus petit, l'"Organisateur" a envoyé une nouvelle version du composant et la version du "Participant" doit être mise à jour. Comme une ou plusieurs mises à jour ont été manquées, le "Participant" DEVRAIT envoyer un message "REFRESH" à l'"Organisateur" pour avoir une version à jour de l'événement.

Dans le cas (2), quelque chose ne va pas. L'"Organisateur" et le "Participant" devraient avoir tous deux les mêmes instances, mais le "Participant" n'a pas l'instance référencée. Dans ce cas, le "Participant" DEVRAIT envoyer un "REFRESH" à l'"Organisateur" pour obtenir une version à jour de l'événement.

Dans le cas (3), les limitations du CUA du "Participant" rendent impossible de correspondre à une instance autre que la seule instance programmée. Dans ce cas, le "Participant" n'a pas besoin d'envoyer de "REFRESH" à l'"Organisateur".

L'exemple ci-dessous montre une séquence dans laquelle un "Participant" envoie un "REFRESH" à l'"Organisateur".

Action	Organisateur	Participant
Mise à jour d'une demande d'instance	"A" envoie un message REQUEST à "B".	
Le participant demande un rafraîchissement parce que le RECURRENT-ID n'a pas été trouvé		"B" envoie un message REFRESH à "A".
Rafraîchir l'événement entier	"A" envoie la dernière copie de l'événement à "B"	
Le participant traite la demande et met à jour l'instance		B" met à jour la dernière copie de la réunion

Demande de "A" :

```
BEGIN:VCALENDAR
METHOD:REQUEST
PRODID:-//Example/ExampleCalendarClient//EN
VERSION:2.0
BEGIN:VEVENT
UID:exemple-12345@exemple.com
SEQUENCE:3
RRULE:FREQ=WEEKLY
RDATE;VALUE=PERIOD:19970819T210000Z/199700819T220000Z
ORGANIZER:mailto:a@exemple.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
DESCRIPTION:IETF-C&S Conference Call
SUMMARY:IETF Calendaring Working Group Meeting
DTSTART:19970801T210000Z
DTEND:19970801T220000Z
RECURRENCE-ID:19970809T210000Z
DTSTAMP:19970726T083000
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

"B" a l'événement avec la propriété "UID" "exemple-12345@exemple.com", mais la valeur de propriété de la "SEQUENCE" de "B" est "1" et l'événement n' a pas une instance à l'heure de récurrence spécifiée. Cela signifie que "B" a manqué au moins une mise à jour et a besoin d'une nouvelle copie de l'événement. "B" demande la dernière copie de l'événement avec le message de rafraîchissement suivant :

```
BEGIN:VCALENDAR
PRODID:-//Example/ExampleCalendarClient//EN
METHOD:REFRESH
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:mailto:a@exemple.com
ATTENDEE:mailto:b@exemple.com
UID:exemple-12345@exemple.com
DTSTAMP:19970603T094000
END:VEVENT
END:VCALENDAR
```

5. Positions de repli du protocole d'application

5.1 Mise en œuvre partielle

Les applications qui prennent en charge la présente spécification ne sont pas obligées de prendre en charge le protocole entier. On décrit ci-après comment les méthodes et propriétés DEVRAIENT "se replier" dans les applications qui ne

prennent pas en charge le protocole complet. Si une méthode ou propriété n'est pas traitée dans cette section, elle peut être ignorée.

5.1.1 Positions de repli relatives aux événements

Méthode	Repli
PUBLISH	Exigé
REQUEST	PUBLISH
REPLY	Exigé
ADD	Exigé si la récurrence est prise en charge ; autrement, repli avec une REQUEST-STATUS "2.8 ; succès, ignore la répétition de l'événement. Programmé comme un seul composant", et programme comme un seul composant.
CANCEL	Exigé
REFRESH	Exigé
COUNTER	Réponse "Non pris en charge".
DECLINECOUNTER	Exigé si COUNTER est mis en œuvre pour les VEVENT; autrement, répond "Non pris en charge"

Propriété iCalendar	Repli
CALSCALE	Ignorer - suppose GREGORIAN.
PRODID	Ignorer
METHOD	Exigé comme décrit dans la liste de Méthode ci-dessus.
VERSION	Ignorer

Composants relatif à un événement	Repli
VALARM	Répondre avec "Non pris en charge".
VTIMEZONE	Exigé si une valeur DateTime se réfère à une zone horaire.

Composant de propriété	Repli
ATTACH	Ignorer
ATTENDEE	Exigé si METHOD est REQUEST; autrement, ignorer.
CATEGORIES	Ignorer
CLASS	Ignorer
COMMENT	Ignorer
COMPLETED	Ignorer
CONTACT	Ignorer
CREATED	Ignorer
DESCRIPTION	Ignorer
DURATION	Exigé
DTSTAMP	Exigé
DTSTART	Exigé
DTEND	Exigé
EXDATE	Ignorer
GEO	Ignorer
LAST-MODIFIED	Ignorer
LOCATION	Exigé
ORGANIZER	Exigé si METHOD est REQUEST ; autrement, ignorer.
PRIORITY	Ignorer
RELATED-TO	Ignorer
RDATE	Ignorer
RRULE	Ignorer - suppose que la première instance se produit sur la propriété DTSTART. Si mis en œuvre, VTIMEZONE DOIT aussi être mis en œuvre.
RECURRENCE-ID	Exigé si RRULE est mis en œuvre ; autrement, ignorer.
REQUEST-STATUS	Exigé
RESOURCES	Ignorer
SEQUENCE	Exigé
STATUS	Ignorer
SUMMARY	Ignorer
TRANSP	Exigé si FREEBUSY est mis en œuvre ; autrement, ignorer.
URL	Ignorer
UID	Exigé
X-	Ignorer

5.1.2 Replis relatifs à Free/Busy

Méthode Repli

PUBLISH	Exigé si les recherches freebusy sont acceptées ; autrement, répond avec un REQUEST-STATUS "3.14; Capacité non prise en charge".
REQUEST	Exigé si les recherches freebusy sont acceptées ; autrement, répond avec un REQUEST-STATUS "3.14; Capacité non prise en charge".
REPLY	Exigé si les recherches freebusy sont acceptées ; autrement, répond avec un REQUEST-STATUS "3.14; Capacité non prise en charge".

Propriété iCalendar Repli

CALSCALE	Ignorer - suppose GREGORIAN.
PRODID	Ignorer
METHOD	Exigé comme décrit dans la liste de Méthode ci-dessus.
VERSION	Ignorer

Composant de propriété Repli

ATTENDEE	Exigé si METHOD est REQUEST; autrement, ignorer.
COMMENT	Ignorer
CONTACT	Ignorer
DTEND	Exigé
DTSTAMP	Exigé
DTSTART	Exigé
DURATION	Ignorer
FREEBUSY	Exigé
ORGANIZER	Exigé si METHOD est REQUEST ; autrement, ignorer.
REQUEST-STATUS	Ignorer
UID	Exigé
URL	Ignorer
X-	Ignorer

5.1.3 Replis relatifs à des choses à faire

Méthode Repli

PUBLISH	Exigé
REQUEST	PUBLISH
REPLY	Exigé
ADD	Exigé si la récurrence est prise en charge ; autrement, repli avec une REQUEST-STATUS "2.8 ; succès, ignore la répétition de l'événement. Programmé comme un seul composant", et programme comme un seul composant.
CANCEL	Exigé
REFRESH	Exigé
COUNTER	Réponse "Non pris en charge".
DECLINECOUNTER	Exigé si COUNTER est mis en œuvre pour les VTODO ; autrement, répond "Non pris en charge".

Propriété iCalendar Repli

CALSCALE	Ignorer - suppose GREGORIAN.
PRODID	Ignorer
METHOD	Exigé comme décrit dans la liste de Méthode ci-dessus.
VERSION	Ignorer

Composants relatifs aux choses à faire Repli

VALARM	Réponse "Non pris en charge".
VTIMEZONE	Exigé si une valeur DateTime se réfère à une zone horaire.

Composant de propriété Repli

ATTACH	Ignorer
ATTENDEE	Exigé si METHOD est REQUEST ; autrement, ignorer.
CATEGORIES	Ignorer
CLASS	Ignorer

COMMENT	Ignorer
COMPLETED	Exigé
CONTACT	Ignorer
CREATED	Ignorer
DESCRIPTION	Exigé si METHOD est REQUEST ; autrement, ignorer.
DUE	Exigé
DURATION	Exigé
DTSTAMP	Exigé
DTSTART	Exigé
EXDATE	Ignorer - répondre avec "Non pris en charge".
LAST-MODIFIED	Ignorer
LOCATION	Ignorer
ORGANIZER	Exigé si METHOD est REQUEST ; autrement, ignorer.
PERCENT-COMPLETE	Ignorer
PRIORITY	Exigé
RECURRENCE-ID	Exigé si RRULE est mis en œuvre ; autrement, ignorer.
RELATED-TO	Ignorer
REQUEST-STATUS	Ignorer
RDATE	Ignorer
RRULE	Ignorer - suppose que la première instance se produit sur la propriété DTSTART. Si mis en œuvre, VTIMEZONE DOIT aussi être mis en œuvre.
RESOURCES	Ignorer
SEQUENCE	Exigé
STATUS	Exigé
SUMMARY	Ignorer
URL	Ignorer
UID	Exigé
X-	Ignorer

5.1.4 Replis relatifs à Journal

Méthode	Repli
PUBLISH	Les mises en œuvre PEUVENT ignorer le type METHOD. Le REQUEST-STATUS "3.14; Capacité non prise en charge" DOIT être retournée.
ADD	Les mises en œuvre PEUVENT ignorer le type METHOD. Le REQUEST-STATUS "3.14; Capacité non prise en charge" DOIT être retournée.
CANCEL	Les mises en œuvre PEUVENT ignorer le type METHOD. Le REQUEST-STATUS "3.14; Capacité non prise en charge" DOIT être retournée.

Propriété iCalendar	Repli
CALSCALE	Ignorer - suppose GREGORIAN.
PRODID	Ignorer
METHOD	Exigé comme décrit dans la liste de Méthode ci-dessus.
VERSION	Ignorer

Composants relatifs à Journal	Repli
VTIMEZONE	Exigé si une valeur DateTime se réfère à une zone horaire.

Composant de propriété	Repli
ATTACH	Ignorer
ATTENDEE	Ignorer
CATEGORIES	Ignorer
CLASS	Ignorer
COMMENT	Ignorer
CONTACT	Ignorer
CREATED	Ignorer
DESCRIPTION	Ignorer
DTSTAMP	Exigé
DTSTART	Exigé
EXDATE	Ignorer
LAST-MODIFIED	Ignorer

ORGANIZER	Ignorer
RECURRENCE-ID	Exigé si RRULE est mis en œuvre ; autrement, ignorer.
RELATED-TO	Ignorer
RDATE	Ignorer
RRULE	Ignorer - suppose que la première instance se produit sur la propriété DTSTART. Si mis en œuvre, VTIMEZONE DOIT aussi être mis en œuvre.
SEQUENCE	Exigé
STATUS	Ignorer
SUMMARY	Exigé
URL	Ignorer
UID	Exigé
X-	Ignorer

5.2 Problèmes de latence

Avec un transport en différé, il est possible que des événements arrivent hors séquence. C'est-à-dire, une méthode "CANCEL" peut être reçue avant de recevoir la "REQUEST" associée pour le composant de calendrier. Ce paragraphe discute quelques un de ces scénarios.

5.2.1 Annulation d'un composant de calendrier inconnu

Quand une méthode "CANCEL" est reçue avant la méthode "REQUEST" originale, le calendrier va être incapable de corréliser la propriété "UID" de l'annulation avec un composant de calendrier existant. Il est suggéré que les messages qui ne peuvent pas être corrélés et qui contiennent aussi des numéros de séquence non zéro soit conservés et non éliminés. Les mises en œuvre PEUVENT les périmier si aucun autre message n'arrive avec la même valeur de propriété "UID" et un numéro de séquence inférieur.

5.2.2 Réponse inattendue d'un délégué inconnu

Quand un "Participant" délègue un élément à un autre CU, il DOIT envoyer une méthode "REPLY" à l'"Organisateur" en utilisant les propriétés "ATTENDEE" pour indiquer que la demande a été déléguée et à qui. Donc, il est possible qu'un "Organisateur" reçoive une "REPLY" d'un CU qui ne figure pas sur sa liste comme un des "Participants" d'origine. La résolution est laissée à la mise en œuvre, mais on s'attend à ce que le logiciel de calendrier accepte la réponse ou la garde jusqu'à ce que la méthode "REPLY" qui s'y rapporte soit reçue du "Déléguant". Si la version de la méthode "REPLY" est périmée, l'"Organisateur" DEVRAIT traiter le message "REFRESH" et mettre à jour le "Délégué" avec la version correcte, pourvu que la délégation à ce délégué soit acceptable.

5.3 Numéro de séquence

Dans certaines conditions, un CUA peut recevoir des demandes et des réponses avec la même valeur de propriété "SEQUENCE". La propriété "DTSTAMP" est utilisée comme départage quand deux éléments avec la même valeur de propriété "SEQUENCE" sont évaluées.

6. Considérations sur la sécurité

iTIP est un protocole de transport abstrait qui va être lié au transport en temps réel, à un transport en différé, et peut-être d'autres transports. Le protocole de transport va être chargé de fournir des facilités d'authentification et de chiffrement en utilisant les mécanismes standard de l'Internet qui sont mutuellement compris entre envoyeur et receveur.

6.1 Menaces sur la sécurité

6.1.1 Usurpation de l'identité de l'organisateur

Dans iTIP, l'"Organisateur" (ou quelqu'un qui travaille au nom de l'"Organisateur") est la seule personne autorisée à faire des changements à un composant de calendrier "VEVENT", "VTODO", ou "VJOURNAL" existant et à le republier ou à redistribuer des mises à jour aux "Participants". Un objet iCalendar qui change ou annule par malveillance un composant de calendrier "VEVENT", "VTODO", ou "VJOURNAL" existant peut être construit par quelqu'un d'autre que l'"Organisateur" et republié ou envoyé aux "Participants".

6.1.2 Usurpation d'identité de participant

Dans iTIP, un "Participant" d'un composant de calendrier "VEVENT" ou "VTODO" (ou quelqu'un qui travaille au nom du "Participant") est la seule personne autorisée pour mettre à jour tout paramètre associé à sa propriété "ATTENDEE" et l'envoyer à l'"Organisateur". Un objet iCalendar qui change par malveillance les paramètres "ATTENDEE" peut être construit par quelqu'un autre que le "Participant" réel et envoyé à l'"Organisateur".

6.1.3 Remplacement non autorisé de l'organisateur

Il va y avoir des circonstances où les "Participants" à un événement ou une chose à faire décident, en utilisant des mécanismes hors bande, que l'"Organisateur" doit être remplacé. Quand le nouvel "Organisateur" envoie le "VEVENT" ou "VTODO" mis à jour, le CUA du "Participant" va détecter que l'"Organisateur" a été changé, mais il n'a pas de moyen de savoir si le changement a fait ou non l'objet d'un accord mutuel.

6.1.4 Espionnage et intégrité des données

L'objet iCalendar est construit avec du texte en clair lisible par l'homme. Toute information contenue dans un objet iCalendar peut être lu et/ou changé par des personnes non autorisées lorsque l'objet est en transit.

6.1.5 Inondation de calendrier

Les mises en œuvre pourraient fournir un moyen pour incorporer automatiquement des méthodes "REQUEST" dans un calendrier. Cela offre l'opportunité d'inonder un calendrier de demandes qui vont effectivement bloquer tous les temps libres du calendrier.

6.1.6 Demandes REFRESH non autorisées

Il est possible qu'un "Organisateur" reçoive une demande "REFRESH" de quelqu'un qui n'est pas un "Participant" d'un événement ou d'une chose à faire. Seuls les "Participants" à un événement ou une chose à faire sont autorisés à recevoir des réponses aux demandes "REFRESH". Répondre à de telles demandes pour quiconque n'est pas un "Participant" peut être un problème de sécurité.

6.2 Recommandations

Pour une application où l'information est sensible ou critique et où le réseau est soumis à une forte probabilité d'attaque, les transactions iTIP DEVRAIENT être chiffrées et authentifiées. Cela aide à atténuer les menaces d'usurpation d'identité, d'espionnage, et de changements malveillants dans le transit.

6.2.1 Sécurisation des transactions iTIP

Les liens de transport iTIP DOIVENT fournir un mécanisme pour permettre l'authentification de l'identité de l'expéditeur ainsi que la confidentialité et l'intégrité des données transmises. Cela permet au receveur d'un objet iCalendar signé de vérifier l'identité de l'expéditeur. Cet expéditeur peut alors être corrélé à une propriété "ATTENDEE" dans l'objet iCalendar. Si la corrélation est faite et si l'expéditeur est autorisé à faire le changement ou mise à jour demandé, alors l'opération peut se poursuivre. Cela permet aussi le chiffrement du message pour empêcher la lecture non autorisée du contenu des messages en transit. Les documents de liens de transport iTIP décrivent ce processus en détails.

6.2.2 Contrôles des mises en œuvre

La menace de remplacement non autorisé de l'"Organisateur" DEVRAIT être atténuée par un système de calendrier qui utilise le présent protocole en fournissant des contrôles ou alertes qui informent les "Utilisateurs de calendrier" de tels changements d'"Organisateur" et leur permettent de décider si la demande devrait ou non être honorée.

La menace d'inondation d'un calendrier DEVRAIT être atténuée par un système de calendrier qui utilise ce protocole en fournissant des contrôles qui peuvent être utilisés pour limiter les sources acceptables pour les transactions iTIP, et peut-être la taille des messages et le volume de trafic, par source.

La menace de demandes "REFRESH" non autorisées DEVRAIT être atténuée par un système de calendrier qui utilise ce protocole en fournissant des contrôles ou alertes qui permettent aux "Utilisateurs de calendrier" de décider si la demande devrait ou non être honorée. Une mise en œuvre PEUT décider de conserver, à des fins d'audit ou historiques, les "Utilisateurs de calendrier" qui ont fait partie d'une liste des "Participants" et qui ont été ensuite non invités. Des contrôles ou alertes similaires devraient aussi être fournis quand une demande "REFRESH" est reçue de ces "Utilisateurs de

calendrier".

6.2.3 Contrôles et filtrage d'accès

Dans de nombreux environnements, il pourrait y avoir des restrictions sur à qui il est permis de programmer, avec qui et à qui sont les délégués permis pour des "Utilisateurs de calendrier" particuliers.

Les liens de transport iTIP DEVRAIENT fournir des mécanismes pour mettre en œuvre des contrôles d'accès ou des filtrages pour s'assurer que les transactions iTIP n'ont lieu qu'entre des "Utilisateurs de calendrier" autorisés. Cela inclurait d'empêcher un "Utilisateur de calendrier" de programmer avec un autre, ou un "Utilisateur de calendrier" de déléguer à un autre.

6.3 Questions de confidentialité

L'"Organisateur" pourrait vouloir empêcher les "Participants" de savoir quels autres "Participants" participent à un événement ou chose à faire. L'"Organisateur" a le choix d'envoyer des messages iTIP avec une liste complète des "Participants" ou d'envoyer des messages iTIP à chaque "Participant" avec seulement ce "Participant" sur la liste des destinataires.

7. Considérations relatives à l'IANA

7.1 Gabarit d'enregistrement pour les valeurs de REQUEST-STATUS

La présente spécification met à jour la [RFC5545] en ajoutant un registre des valeurs de "REQUEST-STATUS" au registre des éléments iCalendar.

Une valeur de "REQUEST-STATUS" est définie en remplissant le gabarit suivant :

Code d'état : code d'état hiérarchique numérique de retour, suivant les règles définie au paragraphe 3.8.8.3 de la [RFC5545].

Description d'état : description textuelle de l'état. Description brève mais claire de l'erreur.

Données d'exception d'état : données d'exception textuelles. Description brève mais claire de ce qui peut apparaître dans ce champ.

Description : décrit la cause sous-jacente de cette valeur de code d'état.

7.2 Ajouts au registre iCalendar METHOD

Le présent document définit les valeurs suivantes pour la propriété iCalendar "METHOD", en utilisant le gabarit des valeurs du paragraphe 8.2.6 de la [RFC5545]. Elles devraient être ajoutées au registre des méthodes défini au paragraphe 8.3.12 de la [RFC5545]:

7.2.1 METHOD:PUBLISH

Valeur : PUBLISH

Objet : valeur "METHOD" iTIP standard.

Conformité : seulement utilisé avec la propriété "METHOD".

Exemples : voir la présente RFC.

7.2.2 METHOD:REQUEST

Valeur : REQUEST

Objet : valeur "METHOD" iTIP standard.

Conformité : seulement utilisé avec la propriété "METHOD".

Exemples : voir la présente RFC.

7.2.3 METHOD:REPLY

Valeur : REPLY

Objet : valeur "METHOD" iTIP standard.

Conformité : seulement utilisé avec la propriété "METHOD".

Exemples : voir la présente RFC.

7.2.4 METHOD:ADD

Valeur : ADD

Objet : valeur "METHOD" iTIP standard.

Conformité : seulement utilisé avec la propriété "METHOD".

Exemples : voir la présente RFC.

7.2.5 METHOD:CANCEL

Valeur : CANCEL

Objet : valeur "METHOD" iTIP standard.

Conformité : seulement utilisé avec la propriété "METHOD".

Exemples : voir la présente RFC.

7.2.6 METHOD:REFRESH

Valeur : REFRESH

Objet : valeur "METHOD" iTIP standard.

Conformité : seulement utilisé avec la propriété "METHOD".

Exemples : voir la présente RFC.

7.2.7 METHOD:COUNTER

Valeur : COUNTER

Objet : valeur "METHOD" iTIP standard.

Conformité : seulement utilisé avec la propriété "METHOD".

Exemples : voir la présente RFC.

7.2.8 METHOD:DECLINECOUNTER

Valeur : DECLINECOUNTER

Objet : valeur "METHOD" iTIP standard.

Conformité : seulement utilisé avec la propriété "METHOD".

Exemples : voir la présente RFC.

7.3 Registre des valeurs de REQUEST-STATUS

De nouvelles valeurs de "REQUEST-STATUS" peuvent être enregistrées en utilisant le procès décrit au paragraphe 8.2.1 de la [RFC5545]. Le tableau suivant est à utiliser pour initialiser le registre de valeurs de "REQUEST-STATUS" .

Code d'état	État	Référence
2.0	Courant	RFC 5546, paragraphe 3.6.1
2.1	Courant	RFC 5546, paragraphe 3.6.2
2.2	Courant	RFC 5546, paragraphe 3.6.3
2.3	Courant	RFC 5546, paragraphe 3.6.4
2.4	Courant	RFC 5546, paragraphe 3.6.5
2.5	Courant	RFC 5546, paragraphe 3.6.6
2.6	Courant	RFC 5546, paragraphe 3.6.7
2.7	Courant	RFC 5546, paragraphe 3.6.8
2.8	Courant	RFC 5546, paragraphe 3.6.9
2.9	Courant	RFC 5546, paragraphe 3.6.10
2.10	Courant	RFC 5546, paragraphe 3.6.11
2.11	Courant	RFC 5546, paragraphe 3.6.12
3.0	Courant	RFC 5546, paragraphe 3.6.13
3.1	Courant	RFC 5546, paragraphe 3.6.14
3.2	Courant	RFC 5546, paragraphe 3.6.15
3.3	Courant	RFC 5546, paragraphe 3.6.16
3.4	Courant	RFC 5546, paragraphe 3.6.17
3.5	Courant	RFC 5546, paragraphe 3.6.18

3.6	Courant	RFC 5546, paragraphe 3.6.19
3.7	Courant	RFC 5546, paragraphe 3.6.20
3.8	Courant	RFC 5546, paragraphe 3.6.21
3.9	Courant	RFC 5546, paragraphe 3.6.22
3.10	Courant	RFC 5546, paragraphe 3.6.23
3.11	Courant	RFC 5546, paragraphe 3.6.24
3.12	Courant	RFC 5546, paragraphe 3.6.25
3.13	Courant	RFC 5546, paragraphe 3.6.26
3.14	Courant	RFC 5546, paragraphe 3.6.27
4.0	Courant	RFC 5546, paragraphe 3.6.28
5.0	Courant	RFC 5546, paragraphe 3.6.29
5.1	Courant	RFC 5546, paragraphe 3.6.30
5.2	Courant	RFC 5546, paragraphe 3.6.31
5.3	Courant	RFC 5546, paragraphe 3.6.32

8. Remerciements

Ceci est une mise à jour du document iTIP original rédigé par S. Silverberg, S. Mansour, F. Dawson, et R. Hopson.

Cette révision a été produite par le groupe de travail Calsify de l'IETF, et plusieurs participants ont fait des contributions importantes à la présente spécification, parmi lesquels Oliver Block, Bernard Desruisseaux, Mike Douglass, Tim Hare, Ciny Joy, Bruce Kahn, Reinhold Kainhofer, Eliot Lear, Jonathan Lennox, Andy Mabbett, Aki Niemi, John W. Noerenberg II, Robert Ransdell, et Caleb Richardson.

9. Références

9.1 Références normatives

- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. (MàJ par [RFC8174](#))
- [RFC2368] P. Hoffman, L. Masinter, J. Zawinski, "[Le schéma d'URL mailto](#)", juillet 1998. (P.S.) (Obsolète, voir la [RFC6068](#))
- [RFC5545] B. Desruisseaux, "[Spécification d'objet central de calendrier](#) et de programmation Internet (iCalendar)", septembre 2009. (Remplace [RFC2446](#), MàJ par [RFC5546](#), [7529](#), [7953](#), [7986](#)) (P. S)

9.2 Référence pour information

- [RFC6047] A. Melnikov, "[Protocole d'interopérabilité iCalendar](#) fondée sur le message (iMIP)", décembre 2010. (Remplace RFC2447) (P.S.)

Appendice A. Différences avec la RFC 2446

A.1 Restrictions changées

La présente spécification définit maintenant les combinaisons permises de codes de "REQUEST-STATUS" quand plusieurs composants iCalendar sont inclus dans un message iTIP.

La présente spécification restreint maintenant "RECURRENCE-ID" à une seule occurrence dans tout composant iCalendar dans un message iTIP, comme exigé par la [RFC5545].

Changement de l'entrée "RECURRENCE-ID" dans le tableau des restrictions de composant de "0+" en "0 ou 1", pour s'aligner sur la [RFC5545].

Changement de l'entrée "FREEBUSY" dans le tableau des restrictions de "VFREEBUSY", "PUBLISH", et "REPLY" de

"1+" en "0+", pour s'aligner sur la [RFC5545].

Changement de la description de "FREEBUSY" dans les tableaux de restriction "VFREEBUSY" et "REPLY" pour indiquer que des gammes différentes de "FBTYPE" NE DOIVENT PAS se chevaucher.

Changement de l'entrée "TZNAME" dans le tableau de restriction "VTIMEZONE" de "0 ou 1" en "0+", pour s'aligner sur la [RFC5545].

Changement de l'entrée "COMMENT" dans le tableau de restriction de composant de "0 ou 1" en "0+", pour s'aligner sur la [RFC5545].

Ajout de l'entrée "ATTENDEE" dans le tableau de restriction de "VALARM" pour correspondre au type d'alarme par message électronique dans la [RFC5545].

Changement de l'entrée "CATEGORIES" dans le tableau de restriction de composant de "0 ou 1" en "0+", pour s'aligner sur la [RFC5545].

Changement de l'entrée "RESOURCES" dans le tableau de restriction de composant de "0 ou 1" en "0+", pour s'aligner sur la [RFC5545].

Changement de l'entrée "CONTACT" dans le tableau de restriction de "VFREEBUSY" de "0+" en "0 ou 1", pour s'aligner sur la [RFC5545].

Changement de l'entrée "UID" dans les tableaux de restriction de "VFREEBUSY" et "PUBLISH" de "0" à "1", pour s'aligner sur la [RFC5545].

Ajout de l'entrée "COMPLETED" dans les tableaux de restriction de "VTODO" pour s'aligner sur la [RFC5545].

Ajout de l'entrée "REQUEST-STATUS" dans les tableaux de restriction de "VJOURNAL" pour s'aligner sur la [RFC5545].

A.2 Caractéristiques déconseillées

La propriété "EXRULE" a été retirée dans la [RFC5545] et ses références ont été supprimées dans le présent document.

La valeur "PROCEDURE" pour la propriété "ACTION" a été supprimée dans la [RFC5545] et ses références ont été supprimées aussi dans le présent document.

L'option "THISANDPRIOR" pour le paramètre "RANGE" a été supprimée dans la [RFC5545] et ses références ont été supprimées aussi dans le présent document.

Adresse de l'éditeur

Cyrus Daboo
Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
USA

mél : cyrus@daboo.name
URI : <http://www.apple.com/>