

---

# draft-armijo-ldap-treedelelete-02

## Contrôle de suppression d'arborescence

Le contrôle est inclu dans un message DelRequest comme partie du champ controls. Le controlType est **1.2.840.113556.1.4.805**, la criticité peut être TRUE ou FALSE, et controlValue est absent.

Ce contrôle permet à un client de supprimer un subtree complet. Cela peut être fait seulement si l'utilisateur authentifié a les permissions pour compléter l'opération. Le serveur doit vérifier pour voir si l'utilisateur authentifié a les permissions appropriées pour supprimer l'objet et tous ses descendants. Ce contrôle doit seulement être utilisé avec un message DelRequest. Un serveur doit ignorer le contrôle s'il est utilisé avec un autre message sauf si la criticité est spécifiée. Dans ce cas, toute l'opération doit échouer avec le resultCode unsupportedCriticalExtension. Les serveur prenant en compte ce contrôle doivent le lister dans supportedControl dans le rootDSE.

## Sémantique

Le demande tree delete peut être traitée comme une simple transaction atomique ou comme une séquence de transaction atomique. Dans le dernier cas, tous les états visible par le client produit par l'opération sont consistant avec l'exécution d'une séquence de demande delete normale. Le serveur ne devrait pas laisser traiter les objets d'une manière qui puisse laisser les objets orphelins.

Si une demande tree delete échoue pour une quelconque raison, il peut être retenté et, si la nouvelle tentative réussit, le résultat est le même que si la demande initiale avait réussi. Le client devrait ré-émettre le delRequest jusqu'à ce que le serveur retourne un succès.

Si un client émet une demande pour déplacer un objet dans ( ou créer un objet dans ) ou hors du subtree, en concurrence avec l'exécution d'une demande tree delete, l'effet est indéfinis. Le serveur n'est pas obligé de bloquer de telles requêtes.

Le serveur ne doit pas poursuivre les références stockées dans l'arborescence. Si l'information sur les référants est stocké dans cet section, ce pointeur est supprimé.

## Messages d'erreur

Quand le contrôle tree delete est invoqué, le serveur doit vérifier pour voir si l'utilisateur authentifié a les permissions appropriées pour supprimer l'objet et toutes ses descendances. Si l'utilisateur n'a pas les permissions appropriées, un insufficientAccessRights(50) est retourné.

Le serveur doit identifier tous les objets à supprimer avant que la suppression commence. Si le serveur a un problème à identifier les objets à supprimer, il peut retourner un operationsError(1). L'opération peut être re-tentée si cette erreur est retournée. Si le serveur n'a pas autorité sur des objets dans le subtree, le serveur devrait retourner unwillingToPerform(53).

Les implémentation de serveurs peuvent avoir d'autres restriction sur quels conteneurs peuvent ou ne peuvent pas utiliser le contrôle Tree Delete. Si vous tentez de supprimer un conteneur qui ne peut pas être supprimé du à une restriction spécifique à la plate-forme, le serveur doit retourner unwillingToPerform(53). Le contrôle Tree Delet ne devrait par être re-tenté sur ce conteneur.

Si la limite du nombre d'objets qui peuvent être supprimés en une opération est atteinte, le serveur devrait retourner adminLimitExceeded(11). Les objets traités jusqu'à ce point de limite devraient être supprimés. la demande DelRequest avec le contrôle Tree Delete devrait être ré-émis jusqu'à ce qu'une réponse success soit retourné par le serveur.