

Consultas CUC SQL para recuentos de mensajes y tamaños de buzón

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Bases de datos](#)

[Tablas](#)

[Consultas SQL](#)

[Lista del recuento total de mensajes con un alias conocido](#)

[Lista de usuarios con recuento total de mensajes](#)

[Lista de usuarios con recuento total de mensajes basado en el primer carácter en alias](#)

[Lista de usuarios con mensajes de bandeja de entrada totales](#)

[Lista de usuarios con mensajes eliminados totales](#)

[Lista de usuarios con total, bandeja de entrada y mensajes eliminados](#)

[Lista de recuento de mensajes de usuario con el tiempo de llegada de mensajes más antiguo](#)

[Enumerar recuento de mensajes de usuario con el tiempo de llegada de mensajes más antiguo y el tamaño de buzón / duración total](#)

[Lista de la bandeja de entrada del usuario y los mensajes eliminados cuentan con el tiempo de llegada del mensaje más antiguo y el tamaño de la casilla de correo / duración total](#)

[Lista del número total de mensajes para todos los buzones](#)

[Lista de un tamaño de buzón de usuario con límites de envío y recepción](#)

[Lista de todos los tamaños de buzón de usuario con límites de envío y recepción](#)

[Lista del tamaño total de todos los buzones combinados](#)

Introducción

Este documento describe cómo obtener el recuento de mensajes y el tamaño de un buzón de correo de usuario con consultas de lenguaje de consulta estructurado (SQL) a través de la CLI. Estos datos también se pueden recuperar con la herramienta [Volcado de datos de usuario](#), desde la página [Herramientas de Cisco Unified Communications](#).

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento de Cisco Unity Connection (CUC).

Componentes Utilizados

La información de este documento se basa en las versiones 8.X y posteriores de CUC, pero esta información podría funcionar también para versiones anteriores.

Bases de datos

Las consultas SQL se forman con los datos de estas bases de datos:

- **UnityDirDB** - Esta base de datos contiene la información relacionada con el usuario.
- **UnityMbxDB1** - Esta base de datos contiene la información del buzón del usuario.

Tablas

Las consultas SQL se forman con los datos de estas vistas. Una *vista* es una tabla que es una combinación de dos o más tablas o los mismos datos en una única tabla.

Estas vistas se utilizan en la base de datos **UnityDirDB**:

- **vw_Mailbox**: esta vista contiene la asignación entre el usuario y el buzón.
- **vw_user**: esta vista contiene la información relacionada con el usuario.

Estas vistas se utilizan en la base de datos **UnityMbxDB1**:

- **vw_message**: Esta vista contiene un elemento de mensaje en el sistema. Esta tabla es simplemente un titular de las propiedades del mensaje.
- **vw_Mailbox**: esta vista contiene un buzón en el sistema que contiene mensajes entrantes. Esta tabla contiene información general sobre el buzón individual, para incluir la forma en que se almacenan los mensajes en el buzón.

Consultas SQL

En esta sección se describen las diversas consultas SQL que puede utilizar en CUC.

Lista del recuento total de mensajes con un alias conocido

Ingrese este comando para obtener una lista del recuento total de mensajes con un alias conocido:

```
admin:run cuc dbquery unitymbxdb1 select count (*) as Messages from vw_message,
```

```
unitydirdb:vw_mailbox, unitydirdb:vw_user where mailboxobjectid in (select mailboxid from vw_mailbox where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid and alias='Anirudh')
```

messages

3

Esta consulta es una consulta de doble base de datos de alta complejidad que involucra varias tablas. En el caso de los servidores con una base de datos muy grande y tamaño de casilla de correo, puede transcurrir un período de tiempo prolongado antes de que aparezca una salida, incluso durante más de una hora, lo que no es ideal. En estos escenarios, puede utilizar esta consulta en su lugar:

```
admin:run cuc dbquery unitymbxdb1 select count (*) as Messages from vw_message where mailboxobjectid in (select mailboxobjectid from vw_mailbox where description='Anirudh')
```

messages

3

La primera consulta devuelve los datos cuando se menciona **el alias**, que es **único**. La segunda consulta devuelve los datos cuando se **menciona la descripción**, que **NO es única**.

Nota: Cuando se crea el buzón, la descripción es la misma que el alias; sin embargo, cuando se actualiza el alias, la descripción no se actualiza. Para las bases de datos pequeñas, la primera consulta es ideal. Para demostrar que la descripción no se cambia después de modificar el alias, **test3** se modifica a **Atest3** y se utiliza para las secciones siguientes.

Lista de usuarios con recuento total de mensajes

Ingrese este comando para obtener una lista de usuarios con el recuento total de mensajes:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages from vw_message, unitydirdb:vw_mailbox, unitydirdb:vw_user where mailboxobjectid in (select mailboxid from vw_mailbox where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) group by alias order by messages desc
```

userid messages

Anirudh 3

Atest3 2

undeliverablemessagesmailbox 1

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as Messages from vw_message, vw_mailbox where vw_mailbox.mailboxobjectid = vw_message.mailboxobjectid group by description order by messages desc
```

description messages

```
-----  
Anirudh 3  
test3 2  
undeliverablemessagesmailbox 1
```

Nota: En la segunda consulta, la descripción no cambia de **test3** a **Atest3** después de cambiar el alias.

Lista de usuarios con recuento total de mensajes basado en el primer carácter en alias

Ingrese este comando para obtener una lista de usuarios con el recuento total de mensajes basado en el primer carácter de un alias:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages from  
vw_message, unitydirdb:vw_mailbox, unitydirdb:vw_user where deleted='0' and  
mailboxobjectid in (select mailboxid from vw_mailbox where unitydirdb:  
vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) and alias like 'A%' group by  
alias order by messages
```

```
userid messages
```

```
-----  
Atest3 2  
Anirudh 3
```

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as Messages from  
vw_message, vw_mailbox where vw_mailbox.mailboxobjectid =  
vw_message.mailboxobjectid and description like 'A%' group by description order  
by messages
```

A continuación se muestran algunas notas importantes sobre esta consulta:

- El **A%** configura la consulta para que se active alias que comienzan con la letra A.
- El formato es **donde columnname como 'condición'**. Aquí, el nombre de la columna es **alias** para la primera consulta y **descripción** para la segunda consulta.

A continuación se muestran algunas condiciones de ejemplo:

- **_n%** - La primera letra puede ser cualquier carácter (comodín), seguida de la letra **n** y cualquier número de caracteres.
- **%s** - Esto configura la consulta para que llegue a los alias que terminan con las letras **s**.

Las consultas que se mencionan hasta ahora se utilizan para obtener el total de mensajes (elementos de bandeja de entrada y eliminados). En la siguiente sección se describen las consultas que se utilizan para obtener el número total de mensajes en la bandeja de entrada y los elementos eliminados.

Lista de usuarios con mensajes de bandeja de entrada totales

Ingrese este comando para obtener una lista de usuarios con el total de mensajes de la bandeja de entrada:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as
inboxmessages from vw_message, unitydirdb:vw_mailbox, unitydirdb:vw_user
where deleted='0' and mailboxobjectid in (select mailboxid from vw_mailbox
where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) and
alias like 'A%' group by alias order by inboxmessages
```

```
userid inboxmessages
-----
```

```
Atest3 2
Anirudh 3
```

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as InboxMessages
from vw_message, vw_mailbox where vw_mailbox.mailboxobjectid =
vw_message.mailboxobjectid and deleted = '0' and description like 'A%' group by
description order by InboxMessages
```

A continuación se muestran algunas notas importantes sobre esta consulta:

- El **A%** configura la consulta para que se active alias que comienzan con la letra A.
- El formato es **donde columnname como 'condición'**. Aquí, el nombre de la columna es **alias** para la primera consulta y **descripción** para la segunda consulta.

A continuación se muestran algunas condiciones de ejemplo:

- **_n%** - La primera letra puede ser cualquier carácter (comodín), seguida de la letra **n** y cualquier número de caracteres.
- **%s** - Esto configura la consulta para que llegue a los alias que terminan con las letras **s**.

Nota: En este ejemplo, se utiliza una condición para limitar a los usuarios con un alias/descripción que comienza con la letra **A**.

Estas son algunas variaciones de esta consulta:

- Para enumerar todos los usuarios, elimine **y alias como 'A%'** en la primera consulta, o **y descripción como 'A%'** en la segunda consulta.
- Para enumerar un usuario concreto (enumere el recuento del buzón de Anirudh, por ejemplo), reemplace **y alias como 'A%'** con **y alias='Anirudh'** en la primera consulta, o reemplace **y la descripción como 'A%'** con **la descripción ='Anirudh'** en la segunda consulta. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.

Lista de usuarios con mensajes eliminados totales

Ingrese este comando para obtener una lista de usuarios con el total de mensajes eliminados:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as
deletedmessages from vw_message, unitydirdb:vw_mailbox, unitydirdb:vw_user
where deleted='1' and mailboxobjectid in (select mailboxid from vw_mailbox
where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) and
alias like 'A%' group by alias order by deletedmessages
```

No records found

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as
deletedmessages from vw_message, vw_mailbox where vw_mailbox.mailboxobjectid=
vw_message.mailboxobjectid and deleted = '1' and description like 'A%' group
by description order by deletedmessages
```

Nota: En este ejemplo no hay mensajes eliminados, por lo que el resultado aparece como **No se encontraron registros**.

A continuación se muestran algunas notas importantes sobre esta consulta:

- El **A%** configura la consulta para que se active alias que comienzan con la letra A.
- El formato es **donde columnname como 'condición'**. Aquí, el nombre de la columna es **alias** para la primera consulta y **descripción** para la segunda consulta.

A continuación se muestran algunas condiciones de ejemplo:

- **_n%** - La primera letra puede ser cualquier carácter (comodín), seguida de la letra **n** y cualquier número de caracteres.
- **%s** - Esto configura la consulta para que llegue a los alias que terminan con las letras **s**.

Nota: En este ejemplo, se utiliza una condición para limitar a los usuarios con un alias/descripción que comienza con la letra **A**.

Estas son algunas variaciones de esta consulta:

- Para enumerar todos los usuarios, elimine **y alias como 'A%'** en la primera consulta, o **y descripción como 'A%'** en la segunda consulta.
- Para enumerar un usuario concreto (enumere el recuento del buzón de Anirudh, por ejemplo), reemplace **y alias como 'A%'** con **y alias='Anirudh'** en la primera consulta, o reemplace **y la descripción como 'A%' con la descripción ='Anirudh'** en la segunda consulta. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.

Lista de usuarios con total, bandeja de entrada y mensajes eliminados

Ingrese este comando para obtener una lista de usuarios con el total, la bandeja de entrada y los mensajes eliminados:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages,
```

```
sum(case when deleted='0' then 1 else 0 end) as Inboxmessages, sum(case when
deleted='1' then 1 else 0 end) as Deletedmessages from vw_message, unitydirdb:
vw_mailbox, unitydirdb:vw_user where mailboxobjectid in (select mailboxid from
vw_mailbox where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid)
group by alias order by messages desc
```

```
userid messages inboxmessages deletedmessages
```

```
-----
```

```
Anirudh 3 3 0
```

```
Atest3 2 2 0
```

```
undeliverablemessagesmailbox 1 1 0
```

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description as UserID, count (*) as
messages, sum(case when deleted='0' then 1 else 0 end) as Inboxmessages, sum
(case when deleted='1' then 1 else 0 end) as Deletedmessages from vw_mailbox
join vw_message on vw_message.mailboxobjectid=vw_mailbox.mailboxobjectid
group by description order by messages desc
```

A continuación se muestran algunas notas importantes sobre esta consulta:

- El **A%** configura la consulta para que se active alias que comienzan con la letra A.
- El formato es **donde columnname como 'condición'**. Aquí, el nombre de la columna es **alias** para la primera consulta y **descripción** para la segunda consulta.

A continuación se muestran algunas condiciones de ejemplo:

- **_n%** - La primera letra puede ser cualquier carácter (comodín), seguida de la letra **n** y cualquier número de caracteres.
- **%s** - Esto configura la consulta para que llegue a los alias que terminan con las letras **s**.

Nota: En este ejemplo, se utiliza una condición para limitar a los usuarios con un alias/descripción que comienza con la letra **A**.

Estas son algunas variaciones de esta consulta:

- Para enumerar todos los usuarios, elimine **y alias como 'A%'** en la primera consulta, o **y descripción como 'A%'** en la segunda consulta.
- Para enumerar un usuario concreto (enumere el recuento del buzón de Anirudh, por ejemplo), reemplace **y alias como 'A%'** con **y alias='Anirudh'** en la primera consulta, o reemplace **y la descripción como 'A%'** con **la descripción ='Anirudh'** en la segunda consulta. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.

Lista de recuento de mensajes de usuario con el tiempo de llegada de mensajes más antiguo

Esta consulta se puede utilizar para determinar si los trabajos programados para la limpieza de mailboxe tienen efecto:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages,
min(arrivaltime) as OldestMessageTime from vw_message, unitydirdb:vw_mailbox,
unitydirdb:vw_user where mailboxobjectid in (select mailboxid from vw_mailbox
where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) group
by alias order by messages desc
```

```
userid messages oldestmessagetime
```

```
-----
Anirudh 3 2013-03-19 14:38:14.459
Atest3 2 2013-01-18 05:49:45.355
undeliverablemessagesmailbox 1 2012-07-05 01:10:19.961
```

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as Messages,
min(arrivaltime) as OldestMessageTime from vw_message, vw_mailbox where
vw_mailbox.mailboxobjectid = vw_message.mailboxobjectid group by description
order by messages desc
```

Nota: Para obtener la Hora de la Última Llegada para las consultas en las secciones anteriores, agregue `min(arrivaltime)` como **TiempoDeMensaje Antiguo** justo después de `contar(*)` como **Mensajes**.

Estas son algunas variaciones de esta consulta:

- Para enumerar todos los usuarios con alias que comienzan con la letra A, agregue **y alias como 'A%'** en la primera consulta justo antes del **grupo por condición de alias**, o **y descripción como 'A%'** en la segunda consulta justo antes del **grupo por condición de descripción**. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.
- Para enumerar un usuario concreto (enumere el recuento del buzón de Anirudh, por ejemplo), agregue **y alias='Anirudh'** en la primera consulta justo antes del **grupo por condición de alias**, o **y descripción ='Anirudh'** en la segunda consulta justo antes de la condición de **descripcióngrupo**. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.

Enumerar recuento de mensajes de usuario con el tiempo de llegada de mensajes más antiguo y el tamaño de buzón / duración total

Ingrese este comando para obtener una lista del conteo de mensajes del usuario con la hora de llegada del mensaje más antigua y el tamaño del buzón de correo (sin duración total):

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages,
min(arrivaltime) as OldestMessageTime, vw_mailbox.bytesize from vw_message,
vw_mailbox, unitydirdb:vw_mailbox, unitydirdb:vw_user where
vw_message.mailboxobjectid=vw_mailbox.mailboxobjectid and
vw_mailbox.mailboxobjectid in (select mailboxid from vw_mailbox where
unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) group by
alias, vw_mailbox.bytesize order by messages desc
```

```
userid messages oldestmessagetime bytesize
```

```
-----
Anirudh 3 2013-03-19 14:38:14.459 93319
```


Nota: Para obtener la duración total de los mensajes: agregue ", **sum(duration/1000) as TotalDuration_In_sec**" justo antes de "**from vw_message**". No olvide la coma antes de la suma. Esto también se puede utilizar para las consultas de las secciones anteriores.

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as Messages,
min(arrivaltime) as OldestMessageTime, vw_mailbox.bytesize from vw_message,
vw_mailbox where vw_mailbox.mailboxobjectid = vw_message.mailboxobjectid
group by description, vw_mailbox.bytesize order by messages desc
```

Nota: Para obtener la duración total de los mensajes: agregue ", **sum(duration/1000) as TotalDuration_In_sec**" justo antes de "**from vw_message**". No olvide la coma antes de la suma. Esto también se puede utilizar para las consultas de las secciones anteriores.

Estas son algunas variaciones de esta consulta:

- Para enumerar todos los usuarios con alias que comienzan con la letra A, agregue **y alias como 'A%'** en la primera consulta justo antes del **grupo por** condición de **alias**, o **y descripción como 'A%'** en la segunda consulta justo antes del **grupo por** condición de **descripción**. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.
- Para enumerar un usuario concreto (enumere el recuento del buzón de Anirudh, por ejemplo), agregue **y alias='Anirudh'** en la primera consulta justo antes del **grupo por** condición de **alias**, o **y descripción ='Anirudh'** en la segunda consulta justo antes de la condición de **descripcióngrupo**. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.

Lista de la bandeja de entrada del usuario y los mensajes eliminados cuentan con el tiempo de llegada del mensaje más antiguo y el tamaño de la casilla de correo / duración total

Ingrese este comando para obtener una lista de la bandeja de entrada del usuario y el conteo de mensajes eliminados con la hora de llegada del mensaje más antigua y el tamaño del buzón (sin duración total):

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as
TotalMessages, sum(case when deleted='0' then 1 else 0 end) as Inbox,
sum(case when deleted='1' then 1 else 0 end) as Deleted, min
(arrivaltime) as OldestMessageTime, vw_mailbox.bytesize from vw_message,
vw_mailbox, unitydirdb:vw_mailbox, unitydirdb:vw_user where
vw_message.mailboxobjectid=vw_mailbox.mailboxobjectid and
vw_mailbox.mailboxobjectid in (select mailboxid from vw_mailbox where
unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) group
by alias, vw_mailbox.bytesize order by TotalMessages desc
```

```
userid total inbox deleted oldestmessagetime byte
messages size
```

```
-----  
Anirudh 3 3 0 2013-03-19 14:38:14.459 93319  
Atest3 2 2 0 2013-01-18 05:49:45.355 59890  
undeliverable 1 1 0 2012-07-05 01:10:19.961 317003  
messagesmailbox
```

Nota: Para obtener la duración total de los mensajes: agregue ", **sum(duration/1000) as TotalDuration_In_sec**" justo antes de "**from vw_message**". No olvide la coma antes de la suma. Esto también se puede utilizar para las consultas de las secciones anteriores.

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as  
TotalMessages, sum(case when deleted='0' then 1 else 0 end) as Inbox,  
sum(case when deleted='1' then 1 else 0 end) as Deleted, min(arrivaltime)  
as OldestMessageTime, vw_mailbox.bytesize from vw_message, vw_mailbox  
where vw_mailbox.mailboxobjectid = vw_message.mailboxobjectid group by  
description, vw_mailbox.bytesize order by TotalMessages desc
```

Nota: Para obtener la duración total de los mensajes: agregue ", **sum(duration/1000) as TotalDuration_In_sec**" justo antes de "**from vw_message**". No olvide la coma antes de la suma. Esto también se puede utilizar para las consultas de las secciones anteriores.

Estas son algunas variaciones de esta consulta:

- Para enumerar todos los usuarios con alias que comienzan con la letra A, agregue **y alias como 'A%'** en la primera consulta justo antes del **grupo por** condición de **alias**, o **y descripción como 'A%'** en la segunda consulta justo antes del **grupo por** condición de **descripción**. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.
- Para enumerar un usuario concreto (enumere el recuento del buzón de Anirudh, por ejemplo), agregue **y alias='Anirudh'** en la primera consulta justo antes del **grupo por** condición de **alias**, o **y descripción ='Anirudh'** en la segunda consulta justo antes de la condición de **descripcióngrupo**. Asegúrese de que el reemplazo se encuentra en la posición exacta o de que la consulta falla.

Lista del número total de mensajes para todos los buzones

Ingrese este comando para obtener el número total de mensajes para todos los buzones combinados:

```
admin:run cuc dbquery unitymbxdb1 select count(*) as messages  
from vw_message
```

```
messages
```

```
-----  
6
```

Lista de un tamaño de buzón de usuario con límites de envío y recepción

Ingrese este comando para obtener el tamaño del buzón de correo del usuario con límites de envío y recepción:

```
admin:run cuc dbquery unitydirdb select alias as UserID,bytesize,send,receive,
warning from vw_user,unitymbxdb1:vw_mailbox where vw_user.objectid in (select
userobjectid from vw_usermailboxmap where
vw_usermailboxmap.mailboxid=unitymbxdb1:vw_mailbox.mailboxobjectid and
alias='Anirudh')
```

```
userid bytesize send receive warning
-----
Anirudh 93319 13000000 14745600 12000000
```

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, bytesize,send,receive,
warning from vw_mailbox where description ='Anirudh'
```

Lista de todos los tamaños de buzón de usuario con límites de envío y recepción

Ingrese este comando para obtener todos los tamaños de casilla de correo del usuario con límites de envío y recepción:

```
admin:run cuc dbquery unitydirdb select alias as UserID,bytesize,send,receive,
warning from vw_user,unitymbxdb1:vw_mailbox where vw_user.objectid in (select
userobjectid from vw_usermailboxmap where
vw_usermailboxmap.mailboxid=unitymbxdb1:vw_mailbox.mailboxobjectid) order by
bytesize desc
```

```
userid bytesize send receive warning
-----
undeliverablemessagesmailbox 317003 13000000 14745600 12000000
Anirudh 93319 13000000 14745600 12000000
Atest3 59890 13000000 14745600 12000000
Solomon 0 13000000 14745600 12000000
UnityConnection 0 50000000 50000000 45000000
Suvir 0 13000000 14745600 12000000
dsas 0 13000000 14745600 12000000
test1 0 13000000 14745600 12000000
Atest2 0 13000000 14745600 12000000
operator 0 13000000 14745600 12000000
```

Por las mismas razones que se mencionan en la sección [Lista del recuento total de mensajes con alias conocido](#), también se puede utilizar esta consulta:

```
admin:run cuc dbquery unitymbxdb1 select description, bytesize,send,receive,
warning from vw_mailbox order by bytesize desc
```

Como una variación de esta consulta para enumerar a todos los usuarios con alias que comienzan con la letra A, añada y alias como 'A%' en la primera consulta justo después de la vw_usermailboxmap.mailboxid=unitymbxdb1:vw_Mailbox.mailboxobjectid y antes) pedido por condición, o puede agregar donde descripción como 'A%' en la segunda consulta justo antes del pedido por condición. Asegúrese de que esto se agrega en la posición correcta o que la consulta

falla.

Lista del tamaño total de todos los buzones combinados

Ingrese este comando para obtener el tamaño total de todos los buzones combinados:

```
admin:run cuc dbquery unitymbxdb1 select sum (bytesize) from vw_mailbox
```

```
(sum)
```

```
-----
```

```
2683210
```

```
admin:
```