

# 消息計數和郵箱大小的CUC SQL查詢

## 目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[資料庫](#)

[表](#)

[SQL查詢](#)

[列出帶有已知別名的消息總數](#)

[列出總郵件數的使用者](#)

[根據別名中的第一個字元列出總消息數的使用者](#)

[列出收件箱郵件總數的使用者](#)

[列出總刪除郵件數的使用者](#)

[列出包含郵件總數、收件箱和已刪除郵件的使用者](#)

[列出具有最早消息到達時間的使用者消息計數](#)

[列出使用者郵件數以及最舊郵件到達時間和郵箱大小/總持續時間](#)

[列出使用者收件箱和已刪除郵件數以及最早郵件到達時間和郵箱大小/總持續時間](#)

[列出所有郵箱的郵件總數](#)

[列出具有傳送和接收限制的使用者郵箱大小](#)

[列出具有傳送和接收限制的所有使用者郵箱大小](#)

[列出所有郵箱的合併總大小](#)

## 簡介

本文檔介紹如何通過CLI使用結構化查詢語言(SQL)查詢獲取使用者郵箱的消息數量和大小。也可以使用[User Data Dump](#) tool從[Cisco Unified Communications Tools](#)頁檢索此資料。

## 必要條件

### 需求

思科建議您瞭解Cisco Unity Connection(CUC)。

### 採用元件

本檔案中的資訊是根據CUC 8.X及更新版本，但此資訊可能適用於舊版本。

## 資料庫

SQL查詢由以下資料庫中的資料形成：

- **UnityDirDB** — 此資料庫包含與使用者相關的資訊。
- **UnityMbxDB1** -此資料庫包含使用者郵箱資訊。

## 表

SQL查詢與這些檢視中的資料一起形成。檢視是兩個或多個表的組合或一個表中的相同資料的表。

這些檢視用於UnityDirDB資料庫中：

- **vw\_mailbox** — 此檢視包含使用者和郵箱之間的對映。
- **vw\_user** -此檢視包含與使用者相關的資訊。

這些檢視用於UnityMbxDB1資料庫：

- **vw\_message** — 此檢視包含系統中的消息項。此表只是消息屬性的持有者。
- **vw\_mailbox** — 此檢視包含系統上保留入站郵件的郵箱。此表包含有關各個郵箱的一般資訊，包括郵箱中儲存郵件的方式。

## SQL查詢

本節介紹可以在CUC中使用的各種SQL查詢。

### 列出帶有已知別名的消息總數

輸入以下命令可獲取具有已知別名的消息總數清單：

```
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
```

此查詢是涉及多個表的高複雜性、雙資料庫查詢。對於具有非常大的資料庫和郵箱大小的伺服器，輸出出現之前可能會經過較長的時間，甚至超過一小時，這是不理想的。在這種情況下，您可以改用以下查詢：

```
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
```

在提到別名時，第一個查詢將返回資料，別名是Unique。第二個查詢在提及description時返回資料，該描述不是Unique。

**附註：**建立郵箱時，說明與別名相同；但是，當別名更新時，說明不會更新。對於小型資料庫，第一個查詢是理想的。為了證明別名更改後說明沒有更改，將test3修改為Atest3，並用於後續部分。

## 列出總郵件數的使用者

輸入以下命令可獲取具有總消息計數的使用者清單：

```
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
```

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

```
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
```

**附註：**在第二個查詢中，別名更改後，說明不會從test3更改為Atest3。

## 根據別名中的第一個字元列出總消息數的使用者

輸入以下命令可獲取具有根據別名的第一個字元計算的總消息數的使用者清單：

```
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
```

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

```
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
```

以下是關於此查詢的一些重要說明：

- **A%**將查詢設定為以字母A開頭的別名。
- 格式是列名 ( 如「condition」 ) 的位置。在這裡，列名是第一個查詢的別名，是第二個查詢的說明。

以下是一些範例條件：

- **\_n%** — 第一個字母可以是任意字元 ( 萬用字元 ) ，後跟字母n和任意字元數。
- **%s** -這將查詢設定為以字母s結尾的別名**匹配**。

使用迄今為止提到的查詢來獲取郵件總數 ( 收件箱和已刪除郵件 ) 。下一部分介紹用於獲取收件箱中的郵件總數和已刪除郵件的查詢。

## 列出收件箱郵件總數的使用者

輸入以下命令可獲取收件箱郵件總數的使用者清單：

```
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
```

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

```
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
```

以下是關於此查詢的一些重要說明：

- **A%**將查詢設定為以字母A開頭的別名。
- 格式是列名 ( 如「condition」 ) 的位置。在這裡，列名是第一個查詢的別名，是第二個查詢的說明。

以下是一些範例條件：

- **\_n%** — 第一個字母可以是任意字元 ( 萬用字元 ) ，後跟字母n和任意字元數。
- **%s** -這將查詢設定為以字母s結尾的別名**匹配**。

**附註：**在此範例中，使用條件以限制別名/說明以字母A開頭的用戶。

以下是此查詢的一些變體：

- 若要列出所有用戶，請刪除第一個查詢中的「A%」之類的別名或第二個查詢中的「A%」這樣的說明。
- 若要列出特定使用者 ( 例如，列出Anirudh郵箱的計數 ) ，請在第一個查詢中將像「A%」和 `alias='Anirudh'` 這樣的替換和別名，或在第二個查詢中將像「A%」這樣的替換和說明替換為 `and description ='Anirudh'`。確保替換項處於正確的位置，或者查詢失敗。

## 列出總刪除郵件數的使用者

輸入以下命令可獲取包含已刪除郵件總數的使用者清單：

```
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

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

```
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
```

**附註：**在此示例中，沒有已刪除的郵件，因此輸出顯示為No records found。

以下是關於此查詢的一些重要說明：

- **A%**將查詢設定為以字母A開頭的別名。
- 格式是列名 ( 如「condition」 ) 的位置。在這裡，列名是第一個查詢的別名，是第二個查詢的說明。

以下是一些範例條件：

- **\_n%** — 第一個字母可以是任意字元 ( 萬用字元 ) ，後跟字母n和任意字元數。
- **%s** -這將查詢設定為以字母s結尾的別名**匹配**。

**附註：**在此範例中，使用條件以限制別名/說明以字母A開頭的用戶。

以下是此查詢的一些變體：

- 若要列出所有用戶，請刪除第一個查詢中的「A%」之類的別名或第二個查詢中的「A%」這樣的說明。
- 若要列出特定使用者（例如，列出Anirudh郵箱的計數），請在第一個查詢中將像「A%」和 alias='Anirudh」這樣的替換和別名，或在第二個查詢中將像「A%」這樣的替換和說明替換為和description='Anirudh'。確保替換項處於正確的位置，或者查詢失敗。

## 列出包含郵件總數、收件箱和已刪除郵件的使用者

輸入以下命令可獲取包含郵件總數、收件箱和已刪除郵件的使用者清單：

```
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
```

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

```
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
```

以下是關於此查詢的一些重要說明：

- A%將查詢設定為以字母A開頭的別名。
- 格式是列名（如「condition」）的位置。在這裡，列名是第一個查詢的別名，是第二個查詢的說明。

以下是一些範例條件：

- \_n% — 第一個字母可以是任意字元（萬用字元），後跟字母n和任意字元數。
- %s -這將查詢設定為以字母s結尾的別名匹配。

**附註：**在此範例中，使用條件以限制別名/說明以字母A開頭的用戶。

以下是此查詢的一些變體：

- 若要列出所有用戶，請刪除第一個查詢中的「A%」之類的別名或第二個查詢中的「A%」這樣的說明。

- 若要列出特定使用者（例如，列出Anirudh郵箱的計數），請在第一個查詢中將像「A%」和 alias='Anirudh」這樣的替換和別名，或在第二個查詢中將像「A%」這樣的替換和說明替換為和description='Anirudh'。確保替換項處於正確的位置，或者查詢失敗。

## 列出具有最早消息到達時間的使用者消息計數

此查詢可用於確定郵件清理的預定作業是否生效：

```
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
```

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

```
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
```

**附註：**為了獲取前面部分中查詢的最後到達時間，請將min(arrivaltime)新增為OldestMessageTime，僅在count(\*)之後作為Messages。

以下是此查詢的一些變體：

- 若要列出別名以字母A開頭的所有用戶，請在第一個查詢中，新增別名如「A%」（緊接group by alias condition），或在第二個查詢中，新增別名如「A%」（緊接group by description條件）之前的描述。確保替換項處於正確的位置，或者查詢失敗。
- 若要列出特定使用者（例如，列出Anirudh郵箱的計數），請在第一個查詢中新增和 alias='Anirudh'（緊接group by alias condition），或在第二個查詢中新增and description='Anirudh'（緊接group by description condition）。確保替換項處於正確的位置，或者查詢失敗。

## 列出使用者郵件數以及最舊郵件到達時間和郵箱大小/總持續時間

輸入以下命令可獲取包含最早郵件的到達時間和郵箱大小（不含總持續時間）的使用者郵件數的清單：

```
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
Atest3 2 2013-01-18 05:49:45.355 59890
undeliverablemessagesmailbox 1 2012-07-05 01:10:19.961 317003
```

**注意:**要獲取消息的總持續時間，請執行以下操作：在「from\_message」之前新增「, sum(duration/1000)as TotalDuration\_In\_sec」。不要忘記求和之前的逗號。這還可以用於前面幾節中的查詢。

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

```
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
```

**附註：**為了獲得消息的總持續時間：在「from\_message」之前新增「, sum(duration/1000)as TotalDuration\_In\_sec」。不要忘記求和之前的逗號。這還可以用於前面幾節中的查詢。

以下是此查詢的一些變體：

- 若要列出別名以字母A開頭的所有用戶，請在第一個查詢中，新增別名如「A%」(緊接group by alias condition)，或在第二個查詢中，新增別名如「A%」(緊接group by description條件)之前的描述。確保替換項處於正確的位置，或者查詢失敗。
- 若要列出特定使用者（例如，列出Anirudh郵箱的計數），請在第一個查詢中新增和 alias='Anirudh'(緊接group by alias condition)，或在第二個查詢中新增and description ='Anirudh'(緊接group by description condition)。確保替換項處於正確的位置，或者查詢失敗。

## 列出使用者收件箱和已刪除郵件數以及最早郵件到達時間和郵箱大小/總持續時間

輸入以下命令可獲取使用者收件箱和已刪除郵件計數的清單，該清單包含最早郵件的到達時間和郵箱大小（無總持續時間）：

```
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
```

附註：若要獲取郵件總持續時間：在「from\_message」之前新增「, sum(duration/1000)as TotalDuration\_In\_sec」。不要忘記求和之前的逗號。這還可以用於前面幾節中的查詢。

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

```
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
```

註：要獲取消息的總持續時間：在「from\_message」之前新增「, sum(duration/1000)as TotalDuration\_In\_sec」。不要忘記求和之前的逗號。這還可以用於前面幾節中的查詢。

以下是此查詢的一些變體：

- 若要列出別名以字母A開頭的所有用戶，請在第一個查詢中，新增別名如「A%」(緊接group by alias condition)，或在第二個查詢中，新增別名如「A%」(緊接group by description條件)之前的描述。確保替換項處於正確的位置，或者查詢失敗。
- 若要列出特定使用者（例如，列出Anirudh郵箱的計數），請在第一個查詢中新增和 alias='Anirudh'(緊接group by alias condition)，或在第二個查詢中新增and description ='Anirudh'(緊接group by description condition)。確保替換項處於正確的位置，或者查詢失敗。

## 列出所有郵箱的郵件總數

輸入以下命令可獲取合併的所有郵箱的郵件總數：

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

```
messages
-----
6
```

## 列出具有傳送和接收限制的使用者郵箱大小

輸入以下命令可取得具有傳送和接收限制的使用者信箱大小：

```
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
```

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

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

## 列出具有傳送和接收限制的所有使用者郵箱大小

輸入以下命令可獲取具有傳送和接收限制的所有使用者郵箱大小：

```
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
```

由於[列出帶有已知別名的消息總數計數](#)部分中提到的相同原因，此查詢也可以使用：

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

作為此查詢的變體，若要列出所有別名以字母A開頭的用戶，請在vw\_usermailboxmap.mailboxid=unitymbxdb1:vw\_mailbox.mailboxobjectid and before) order by 條件之後在第一個查詢中新增和別名（如「A%」），或者可以在description（如「A%」）後在第二個查詢中新增order by 條件。請確保將其新增到正確的位置，否則查詢失敗。

## 列出所有郵箱的合併總大小

輸入以下命令可獲取合併的所有郵箱的總大小：

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

```
(sum)
-----
2683210
```

```
admin:
```

## 關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。