# User Creation in PEBBLES

# V 0.2

| **Version Number** | **Sections changed** | **Reason** |
| --- | --- | --- |
| 0.1 | Initial Version |  |
| 0.2 | Minimal Dataset Requirements,  Example SQL script to generate dataset | Add mobile /telephone number |

Contents

[User Creation in PEBBLES 1](#_Toc507676078)

[Overview of User Creation in PEBBLEs 2](#_Toc507676079)

[Tables to be populated 2](#_Toc507676080)

[Source data 2](#_Toc507676081)

[Process 2](#_Toc507676082)

[Minimal dataset requirements 3](#_Toc507676083)

[Example SQL script to generate dataset 3](#_Toc507676084)

# Overview of User Creation in PEBBLEs

Currently each user in QL is manually created, which is time consuming & presents issues with data accuracy. As part of the data migration and hierarchy work it has been identified that it would be possible to automatically create all users in QL, based on AD user accounts. The basic user creation process would work as follows: -

* A QL user account is created for all AD users without a QL user account, and linked to the AD account
* The user account is given a ‘BLANK’ default workgroup that does not give any QL permissions
* These users are also set up as Responsible Persons in QL
* The cross reference between user and Responsible Person is created

When the ‘hierarchy’ integration with ResourceLink runs the following is then completed: -

* The user is given the correct workgroups and default workgroup as defined in ResourceLink
* The user is entered as Responsible Person as defined in ResourceLink
* The user’s line manager is populated

The process will be run both as part of initial migration, and on a nightly basis thereafter

# Tables to be populated

When creating a user in PEBBLEs the following tables are required to be populated : -

|  |  |
| --- | --- |
| **Table Name** | **Description of Contents** |
| menuser | Main user table. |
| menldapuser | Link between QL username and AD user |
| menugrp | List of all workgroups for a user |
| cmppers | Responsible Persons |
| hmgrpxrf | Cross reference between menuser and cmppers. Also holds line manager |

# Pre-requisites

A new workgroup called ‘BLANK’ needs to be set up in QL

# Source data

AD Users and start / end dates can be found in [H21\_STAGING].[dbo].[REC\_RESOURCELINK\_ACTIVE\_DIRECTORY]

# Process

1. Identify all active AD users that are not present in menldap user, and either currently in post, or are due to start within 5 days according to ResourceLink. Note that due to posts being included then some users may appear twice in [H21\_STAGING].[dbo].[REC\_RESOURCELINK\_ACTIVE\_DIRECTORY], so a select distinct is required.
2. Choose a suitable QL username. Normal format is Surname + Initial, however to prevent duplicates a number is added, also the total length is 10 characters. For simplicity we have chosen to take the left 7 from the surname, the first letter of the firstname, and then add up to two digits as appropriate. The sample SQL uses cross join and ranking (Row Number) functions to identify what username to use.
3. Populate all 5 tables above with a minimal dataset

# Minimal dataset requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | **Field** | **Value** | **Notes** |
| menuser | [user\_id] | <QL\_USER> |  |
| menuser | [u\_version] | ‘$’ | Fixed |
| menuser | [name] | <EMPLOYEE\_NAME> |  |
| menuser | [group\_id] | ‘BLANK’ | The word ‘blank’ |
| menuser | [sys\_admin] | ‘0’ | Fixed |
| menuser | [accountlocked] | ‘0’ | Fixed |
| menuser | [rec\_acp\_email] | ‘0’ | Fixed |
| menuser | [email\_address] | <EMPLOYEE\_EMAIL\_ADDRESS> |  |
| menuser | [tel\_no] | <COALESCE([EMPLOYEE\_MOBILE\_TELEPHONE\_NUMBER], [EMPLOYEE\_TELEPHONE\_NUMBER])> |  |
| menldapuser | [ldap\_user] | <LDAP\_USER> |  |
| menldapuser | [u\_version] | ‘$’ | Fixed |
| menldapuser | [ldap\_server] | ‘S1VLDOMDC01’ | Fixed |
| menldapuser | [ql\_user] | <QL\_USER> |  |
| menldapuser | [use\_ldap] | ‘1’ | Fixed |
| menldapuser | [use\_ssl] | ‘0’ | Fixed |
| menldapuser | [use\_sso] | ‘1’ | Fixed |
| menugrp | [user\_id] | <QL\_USER> |  |
| menugrp | [group\_id] | ‘BLANK’ | The word ‘blank’ |
| cmppers | [comp\_id] | ‘H21’ | Fixed |
| cmppers | [person\_id] | <QL\_USER> |  |
| cmppers | [u\_version] | ‘!’ | Fixed |
| cmppers | [name] | <EMPLOYEE\_NAME> |  |
| hgmrpxrf | [comp\_id] | ‘H21’ | Fixed |
| hgmrpxrf | [resp\_pers] | <QL\_USER> |  |
| hgmrpxrf | [u\_version] | ‘$’ | Fixed |
| hgmrpxrf | [user\_id] | <QL\_USER> |  |
| hgmrpxrf | [spare\_bool] | ‘0’ | Fixed |

# Example SQL script to generate dataset

Note that for ease of PoC this is based on the RAW\_MENLDAPUSER QL table in the DW

IF OBJECT\_ID('TEMPDB..#E1') IS NOT NULL

DROP TABLE #E1

IF OBJECT\_ID('TEMPDB..#E2') IS NOT NULL

DROP TABLE #E2

IF OBJECT\_ID('TEMPDB..#T1') IS NOT NULL

DROP TABLE #T1

IF OBJECT\_ID('TEMPDB..#T2') IS NOT NULL

DROP TABLE #T2

-- Identify new users to create, build employee name and first part of QL username [QL\_INIT\_USER]

SELECT DISTINCT

[EMPLOYEE\_FIRST\_NAME],

[EMPLOYEE\_LAST\_NAME],

[EMPLOYEE\_FIRST\_NAME] + ' ' + [EMPLOYEE\_LAST\_NAME] AS 'EMPLOYEE\_NAME',

LEFT([EMPLOYEE\_LAST\_NAME],7) + LEFT([EMPLOYEE\_FIRST\_NAME],1) AS 'QL\_INIT\_USER',

[EMPLOYEE\_USER\_NAME],

[EMPLOYEE\_EMAIL\_ADDRESS],

COALESCE([EMPLOYEE\_MOBILE\_TELEPHONE\_NUMBER],[EMPLOYEE\_TELEPHONE\_NUMBER]) AS TEL\_NO

INTO #E1

FROM [H21\_STAGING].[dbo].[REC\_RESOURCELINK\_ACTIVE\_DIRECTORY] D

LEFT JOIN RAW\_MENLDAPUSER Q

ON Q.ldap\_user = D.EMPLOYEE\_USER\_NAME

WHERE [EMPLOYEE\_START\_DATE] < getdate()+5

AND [EMPLOYEE\_END\_DATE] > getdate()

AND [EMPLOYEE\_POST\_START\_DATE] < getdate()+5

AND [EMPLOYEE\_POST\_END\_DATE] > getdate()

AND [EMPLOYEE\_USER\_NAME] is not null

AND Q.[ldap\_user] is null

-- Rank these users to identify any duplicate QL\_INIT\_USER within this dataset

SELECT [EMPLOYEE\_USER\_NAME],

[EMPLOYEE\_FIRST\_NAME],

[EMPLOYEE\_LAST\_NAME],

[EMPLOYEE\_NAME],

[QL\_INIT\_USER],

[EMPLOYEE\_EMAIL\_ADDRESS],

[TEL\_NO],

ROW\_NUMBER() OVER(

PARTITION BY [QL\_INIT\_USER]

ORDER BY [EMPLOYEE\_USER\_NAME] ASC

) AS 'EMP\_ROW\_NUM'

INTO #E2

FROM #E1

-- Set up a table of username endings to cross join with QL\_INIT\_USER to create username options

CREATE TABLE #T1

(

[level\_code] CHAR(2)

)

INSERT INTO #T1

SELECT [level\_code]

FROM (

VALUES

(' '),

('1'),

('2'),

('3'),

('4'),

('5'),

('6'),

('7'),

('8'),

('9')

) sub ([level\_code]);

-- Cross join first part of QL username with username endings to create username options QL\_USER\_OPT

SELECT [EMPLOYEE\_USER\_NAME],

[EMPLOYEE\_FIRST\_NAME],

[EMPLOYEE\_LAST\_NAME],

[EMPLOYEE\_NAME],

[EMPLOYEE\_EMAIL\_ADDRESS],

[TEL\_NO],

[EMP\_ROW\_NUM],

[QL\_INIT\_USER] + [level\_code] AS 'QL\_USER\_OPT'

INTO #T2

FROM #E2, #T1

/\*Remove QL\_USER\_OPT that already exist in menldapuser.

Rank those available, assign to new users QL\_USER\_OPT according to the rank of QL\_INIT\_USER

\*/

SELECT [EMPLOYEE\_USER\_NAME] AS [LDAP\_USER],

[QL\_USER\_OPT] AS [QL\_USER],

[EMPLOYEE\_NAME],

[EMPLOYEE\_EMAIL\_ADDRESS]

FROM(

SELECT

[EMPLOYEE\_USER\_NAME],

[EMPLOYEE\_NAME],

[EMP\_ROW\_NUM],

[QL\_USER\_OPT],

[EMPLOYEE\_EMAIL\_ADDRESS],

[TEL\_NO],

ROW\_NUMBER() OVER(

PARTITION BY [EMPLOYEE\_USER\_NAME]

ORDER BY [QL\_USER\_OPT] ASC

) AS 'ROW\_NUM'

FROM #T2

LEFT JOIN RAW\_MENLDAPUSER Q

ON Q.[QL\_USER] = #T2.[QL\_USER\_OPT]

WHERE Q.[QL\_USER] is null) U

WHERE U.[ROW\_NUM] = [EMP\_ROW\_NUM]

ORDER BY EMPLOYEE\_USER\_NAME