**Guide to checking the files created by the PMI Consolidation Software**

The output of the PMI Consolidation Software is a set of report files. All the files that need checking have been created in a format that makes it possible to make simple Yes/No decisions (Yes it matches/No it doesn’t match) or simple Yes/Yes decisions (Yes, this **secondary** PMI patient should be linked to Yes, this specific master **PMI** patient). In Phase 1 only Yes/No decisions need to be made. In Phase 2, a Yes/Yes decision or a No decision needs to be made for every **secondary** PMI record.

These files require you to enter “Y” or “N” and nothing else (except for the “secondary”\_NotFound.xlsx file, which is a special case described below). These files are post processed, so entering anything else, such as “DECEASED”, may confuse the post processing software and may even prevent it from running correctly. For instance, during processing Phase 1 files, you may find the correct **master** PMI **UR** number for a patient. Unfortunately, there is no way of recording this **master** PMI **UR** number in these files, nor making use of that information. This is a safety feature, as typed information is subject to typing errors, such as leaving out a digit, or swapping two digits. You can of course record this information elsewhere as this **master** PMI **UR** number is likely to be thrown up as an alternative in Phase 2 and you’ll be able to select it by typing “Y” against the matching **master** PMI record.

**Phase 1 – assumes the altUR number is correct, but the other details may be wrong**

1. All these files require a simple Yes/No decision (Yes, this **secondary** PMI record is a good enough match to the following **master** PMI record, or No, it isn’t an acceptable match and the PMI Consolidation Software needs to look for a better match).
2. All these files contain either “\_mismatch\_” or “\_ur\_” in the filename. The “\_ur\_matched\_alias” and “\_ur\_matched\_merged” are reports of patients where the AltUR number is not up to date and could/should be updated. However, these patients are also in the “\_ur\_matched\_n” files, which is where you should mark them as “Y’ - match, or “N” – not a match. If you mark them as a match (“Y”), then you will need to ensure that the AltUR gets updated to the correct, current UR. [**Note:** the “matched.py” program, which you run after you have checked all the files, will create a file called “fix.xlsx” which contains all the AltUR’s that need to be updated and the matching current UR]
3. Start with all the “\_mismatch\_ToDo” files. [The “\_mismatch\_Done” files are a by-product of re-running the **Phase 1** process, after some mismatches have been confirmed as mismatches]
4. Type a “Y” or “N” (lowercase will do) in the “Checked” column (which is the first column) on the same line as the **secondary** PMI patient information (which is the first line in each set and contains the message about the mismatch). Normally, this decision can be made based upon nothing more than the patient information in these files (i.e do the patients share the same address, medicare number etc). It is normal to take a conservative approach and type “N” if in doubt. Typing “N” means that this **secondary** PMI record will be processed again in Phase 2, which may find the same match, but it may also find other “better” matches. Marginally ambiguous patient should be checked manually; “Jack Smith” doesn’t match “Jock Smith but if their addresses match then further investigation my may warranted. Phase 2 could recommend a match of “Jack Smith” to a “Jack Smith” with the same date of birth, but a different UR and a completely different address as a “better match”, but is it?
5. If extensive checking (probabilistic matching) has been requested (-E option) then two additional reports will be created (“\_Extensive\_Matches” and “\_Probable\_Matches”). These reports contain additional information about the “matched” records and should be referenced when checking the mismatches.
6. It is unusual to check the “\_ur\_undefined” file. It would only be relevant if the **secondary** PMI extract was created well after the **master** PMI extract or if there was potentially a problem with the **master** PMI extract. However, it can be checked, and you can over write the “not found in” message in column 1 with a “Y”, against any patient that is found in the **master** application, with the same **UR** number as the **secondary** PMI **altUR** number. However, getting a more up to date extract is a better strategy.
7. It is extremely unusual to check the “\_ur\_matched\_n” file(s). Firstly, this should be the majority of patients (hopefully 100%). Secondly, they have all been matched on **UR**=**altUR**, surname, firstname, date of birth and sex. You would only check these in bizarre cases, such as where you had to clean up the first name to an initial or neither extract contained birth date or sex codes. You would check it if there were any records in the “\_ur\_matched\_alias” or “\_ur\_matched\_merged” files which you deemed were not matches. If you can type “N” in the first column, against the **secondary** PMI record, then the associated secondary **PMI** record will be deemed to be unmatched and will get included in the processing in Phase 2.

The Python script **matched.py** will process all these report files and create three files.

* **matched.xlsx** and **notMatched.xlsx** which will inform **Phase 2** and, if necessary, any further rounds of **Phase 1**.
* **fix.xlsx** which is a list of the AltUR’s that were matched to aliases or merged patients and the current UR for this patient. These AltUR’s are in the **matched.xlsx** file.However, these AltUR’s will need to be updated to the current UR for integration to work. This file will constitute part of the recommended edits to the secondary **PMI**.

After this you should re-run matchAltUR.py which will create some new “\_ToDo.xlsx” and “\_Done.xlsx” files. You should check that all the “\_ToDo.xlsx” files are empty; this is everything has moved to the “\_Done.xlsx” files.

**Phase 2 – assumes that the other details may be correct, but the altUR number is definitely wrong**

1. All these files contain “Found” in the filename.
2. Start with the “\_SimilarFound\*\_ToDo” files. [The “\_SimilarFound\*\_Done” files are a by-product of re-running the **Phase 2** process, after some of the suggested ‘found’ matches have been confirmed as correct or discounted as incorrect]
3. These files will contain “sets” of records, consisting of one **secondary** PMI record followed by one **master** PMI records. Scan the subsequent **master** PMI record to see if it is a match for the **secondary** PMI information. You may need to scan alternate sources, such as the **secondary** and **master** applications themselves, and possibly the medical record. If you find a match, type a “Y” (lower case will do) in the “Checked” column against the **secondary** PMI record. If the subsequent **master** PMI record is not a match type a “N” (lowercase will do) against the **secondary** PMI record in the “Checked” column. Every **secondary** PMI record must be marked with a “Y” or an “N”.
4. Then check the “\_DuplicateFound\*\_ToDo” file. [The “\_DuplicateFound\*\_Done” file is a by‑product of re-running the **Phase 2** process, after some of the suggested ‘found’ matches have been confirmed as correct or discounted as incorrect]. This file contains multiple exact matches indicating duplicates in the **master** PMI and requires complex Yes/Yes decisions (“Yes”, the **secondary** PMI record is a match to this specific “Yes” **master** PMI record) or a No decision (“No” - none of the listed **master** PMI records are a match for this **secondary** PMI record). If there is a match then what you are selecting it the “correct” **master** PMI record and a clean up of the **master** PMI may be required.
5. This file will contain “sets” of records, consisting of one **secondary** PMI record followed by two or more **master** PMI records. Scan the subsequent **master** PMI records, looking for a match for the **secondary** PMI information. You may need to scan alternate sources, such as the **secondary** and **master** applications themselves, and possibly the medical record. If you find a match, type a “Y” (lower case will do) in the “Checked” column against the matching **master** PMI record, then go back and type a “Y” (lower case will do) against the **secondary** PMI record in the “Checked” column. If you don’t find a match, go back and type a “N” (lowercase will do) against the **secondary** PMI record in the “Checked” column. Every **secondary** PMI record must be marked with a “Y” or an “N”. If the **secondary** PMI record is marked with a “Y” then one and only of the subsequent **master** PMI records must also be marked with a “Y”.
6. Then check the “\_DuplicateSimilarFound\*\_ToDo” file. [The “\_DuplicateSimilarFound\*\_Done” file is a by‑product of re-running the **Phase 2** process, after some of the suggested ‘found’ matches have been confirmed as correct or discounted as incorrect]. This file contains multiple potential matches and requires complex Yes/Yes decisions (“Yes”, the **secondary** PMI record is a good enough match to this specific “Yes” **master** PMI record) or a No decision (“No” - none of the listed **master** PMI records are a match for this **secondary** PMI record).
7. This file will contain “sets” of records, consisting of one **secondary** PMI record followed by two or more **master** PMI records. Scan the subsequent **master** PMI records, looking for a match for the **secondary** PMI information. You may need to scan alternate sources, such as the **secondary** and **master** applications themselves, and possibly the medical record. If you find a match, type a “Y” (lower case will do) in the “Checked” column against the matching **master** PMI record, then go back and type a “Y” (lower case will do) against the **secondary** PMI record in the “Checked” column. If you don’t find a match, go back and type a “N” (lowercase will do) against the **secondary** PMI record in the “Checked” column. Every **secondary** PMI record must be marked with a “Y” or an “N”. If the **secondary** PMI record is marked with a “Y” then one and only of the subsequent **master** PMI records must also be marked with a “Y”.
8. If extensive checking (probabilistic matching) has been requested (-E option) then two additional reports will be created (“\_Extensive\_Finds” and “\_Probable\_Finds”). These reports contain additional information about the “found” records and should be referenced when checking what was found.
9. The “\_FoundAlias.xlsx” and “\_FoundMerged.xlsx” are reports of patient’s matched to an alias, or a merged record respectively. However, this information is also reported in the “\_SimilarFound\*”, “\_DuplicateFound\*” and “\_DuplicateSimilarFound\*” files. Any found matches that are matches to an alias or a merged record, which are marked as matched (“Y”) will be added to the “fix.xlsx” file, which is a list of AltUR’s which need to be updated and the current UR value that must replace the AltUR for integration to function successfully.
10. Scan the “\_Found\_ToDo\_n” files. [The “\_Found\_Done\_n” files are a by‑product of re-running the **Phase 2** process, after some of the suggested ‘found’ matches have been confirmed as correct or discounted as incorrect]. The PMI Consolidation Software needs confirmation that the matches it has found are, in fact, correct matches. If any found match is in fact not a match, type “N” in the first column against the **secondary** PMI record. The PMI Consolidation Software assumes, for this file only, that nothing is the same as “Y”.
11. Scan the “\_NotFound\_ToDo” file. [The “\_NotFound\_Done” file is a by‑product of re-running the **Phase 2** process, after some of the suggested ‘found’ matches have been confirmed as correct or discounted as incorrect]. It is possible that some of these patients actually exist in the **master** PMI but are so mangled in the **secondary** PMI file as to be hardly recognizable. In fact, so mangled that the PMI Consolidation Software couldn’t recognize them. If you can find them in the **master** application, and you know the correct **UR** number, then you can type it into the first column, against the matching **secondary** PMI record. However, this has to be done **very carefully** as this information is not checked. Any typing mistake at this point will create a corruption in the **secondary** PMI after all recommended edits to the **secondary** PMI have been implemented. The **UR** number that you type here will become a recommended edit for the **secondary** PMI.

The Python script **found.py** will process all these report files and create three files.

* **found.xlsx** contains pairs of PID value and UR value (one pair per row). Each row represents the correct AltUR value for the associated **secondary** PMI record (PID). These **secondary** PMI records will need to be updated and the AltUR value replaced with this UR value.
* **notFound.xlsx** contains PID values for all the **secondary** PMI records for which no match could be found in the **master** PMI records. If these **secondary** PMI records have an AltUR value, then it will need to be removed before integration is enabled. These patient can be added to the **master** PMI which will allocate them a UR value. The matching **secondary** PMI records would then need to be updated with this new UR as their AltUR value.
* **patch.xlsx** contains pairs of PID value and UR value (one pair per row). Each row represents the correct, current UR value for the associated **secondary** PMI record (PID) which has been found to match a **master** PMI record. However, the matching **master** PMI record is an alias, or has been merged to another **master** PMI record. Here the UR value is the correct, current value. These **secondary** PMI records will need to be updated and the AltUR value set to this UR value.