

GOLDBOX'S MERGE-PURGE: STRATEGY, TACTICS AND CHECKLIST

Before there was a Merge/Purge (M/P) in GoldMine, there was a Merge-Purge (M-P) in GoldBox. While the two processes have similar objectives, they operate quite differently. In this document, I will discuss how and why GoldBox's M-P works the way it does, and how you can use it to greatly improve the quality of your GoldMine data.

STRATEGY – GoldMine was originally developed as a Contact Manager, and later transformed into a Customer Relationship Manager (CRM). So it makes sense that the **Main Contact record** was originally intended to represent **one person**. Later, Record Typing made it possible to use the Main Contact record to also represent other things (such as real estate parcels); so the original focus of **one Main Contact record = one person** has been diffused a bit. However, the Main Contact record should still represent **one unique entity**, whether that entity is a person or not.

When two or more Main Contact records exist in a database for the same unique entity, a problem exists...but that problem often remains hidden. For openers, **how do dupe Main Contact records get created?** Mostly, it's by poorly designed Contact imports; by syncing Users who cannot know what records have just been created by others; or just the creation of Contact records by Users who have not done adequate searches.

And **how do duplicates manage to remain hidden?** There are over a dozen indexes on the Main Contact record in GoldMine; so there are many ways to look up records. If there are two Main Contact records for the same person, it's possible that one kind of lookup will find one of the records; while a different kind of lookup will find the other. This can cause various bits of information about the same person to become stored in different Main Contact records, and that can cause GoldMine to retrieve inconsistent information about that person.

Obviously, inconsistent information is not what we want from a CRM. The solution to the problem is the Merge-Purge. It finds duplicate Main Contact records, and merges the information from them into a single record, deleting any excess records that previously existed. If done properly, only true duplicates are affected, and no information is lost.

Here's the strategy of GoldBox's Merge-Purge, which I believe is the best available:

1. **Build a filtered matching index of the database.** This may require careful thought in construction of the matching expression, as well as of the filter or query. Example: use an expression based on the Contact name and Company name to build the matching index; filtered so that no records where either of these fields are empty will be included. This approach works, with the assistance of GoldBox's Smoothing functions, which we'll discuss later, under Tactics.
2. **Use the matching index to segregate only duplicate records.** Duplicates found to have the same matching expression value are gathered into dupe groups.
3. **For each such group, apply an expression that will determine which record within the dupe group will become the Survivor.**
4. **Transfer data from the records that will be deleted to the Survivor record.** This transfer can be custom Backfilling from Main Contact fields to Main Contact fields; it can also be copying attached records from the to-be-deleted Main Contacts to the Survivor Main Contact record. Any differences in Main Contact fields will also be copied to the Notes of a special record stored under the Contacts tab of the Survivor record, to ensure no loss of data.
5. **Delete all records in each dupe group, except for the Survivor record.**

Note that there is no mention in the above list of interrupting the "run" of the Merge-Purge for manual intervention. It is presented as a continuous batch process, because that's exactly what it is. In fact, GoldBox even refers to it as **Batch Mode**.

When we run Merge-Purge in Batch Mode, we are making an **extremely important assumption**: we assume the matches that will be found will **always** be “true matches”... that is, for the same person or entity.

Why is this assumption so important? Consider:

- What could happen if two records for two different people were to be erroneously found to be dupes by the Merge-Purge. The M-P process would combine them into a single record, losing the uniqueness of **BOTH** Contacts. One would be lost entirely (deleted); while the Survivor would be “polluted” with data from the deleted record. Ultimately, two customers could be compromised, even lost, due to bad or missing GoldMine data.

- **Merge-Purge cannot be undone**, except by restoring the entire database from backup. If restoration is not done immediately, recent data will be at risk.

So it's critical that all our matches be “true matches”, when running in Batch Mode. Normally, this means that if we set up a Merge-Purge and run it in Batch Mode, it should be using a **SAFE** matching expression; one which will **never** find false dupes. But what if we want to try a matching expression in which we don't have quite that much confidence?

GoldBox has an option for that situation, called **Count Mode with Preview**. It completes items 1 – 3 on page 1; then stops, after populating the **Merge-Purge Primer Table**. You then “work” the Primer Table. When done, you reset the M-P to Batch Mode and run it.

The Primer Table is a preview of what will happen when the Merge-Purge is run in Batch Mode; but it's much more than that. It's called a “Primer” because it offers options to alter the default operation of Batch Mode, group by group. In effect, it allows us to use less safe matching, **PROVIDED** that we are willing to use the Primer Table to look at each dupe group found, and to **eliminate** the dupe groups (or specific records within dupe groups) that are NOT true matches. Once all these errors are corrected, the M-P can then be run in Batch Mode, because all the matches found will be completely safe.

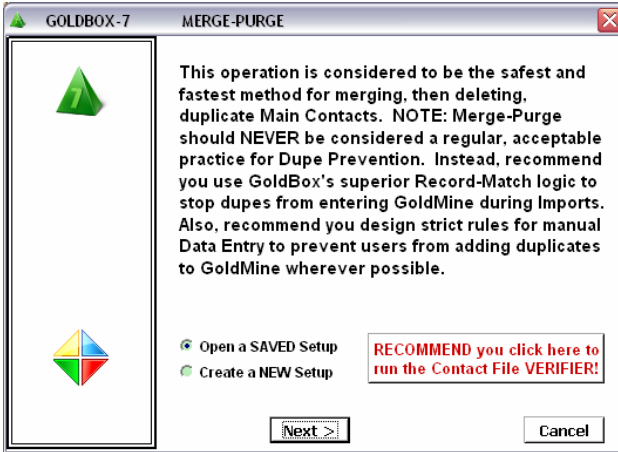
There is also a separate **Manual Mode** of GoldBox's Merge-Purge. But, because I never use it, I won't discuss it in this document.

In summary, the **strategy of Merge-Purge** is:

- A. Using one or more **SAFE** match Batch Mode setups, identify as many true matches as possible, to quickly merge them.
- B. Using one (or more) less safe matching schemes in conjunction with the Primer Table, find matches that cannot be found using SAFE matching alone. For each, correct any erroneous matching in the Primer Table, then run as Batch Mode.

In essence, this process supports what I call the “**Golden Rule of matching**”: **it is much better to fail to identify two matching records than to identify two records for different people (or entities) as a match.** One error can be corrected later; the other cannot.

TACTICS – Filling out GoldBox's Merge-Purge Setup forms is relatively simple:



Part of the process of creating the setup is this page; note the comments, which are important. But the most critical thing here is the red-lettered button that opens the **Contact File VERIFIER**. The **Checklist** that's included in this document specifies how to satisfy this recommendation.

The Checklist also makes it clear that it's common to use multiple Merge-Purge setups to complete the job. Usually, all but one of them will have matching expressions/filters that are **SAFE**. In such setups, the matching criteria for each of these M-Ps is so restrictive that it is simply

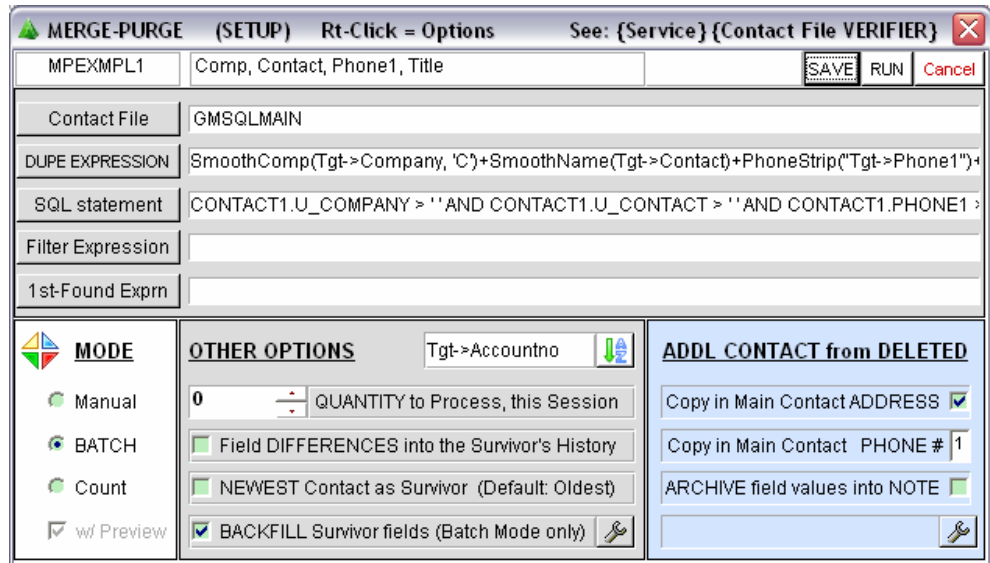
beyond practical possibility that incorrect matches will be found. An example of such criteria might be **Company + Contact + Phone1 + Title**. It's hard to imagine two different people where all four of those values would be exactly the same.

On the other hand, it's easy to imagine that the same person could have two Main Contact records in GoldMine; but that one of these four values would be different. The Title field might be empty in one, for example. Actually, if **ANY** match field is empty in either of two dupes, that would be enough to cause the match to fail, because our SQL statement or filter would eliminate that record from even being considered by the Merge-Purge. This is exactly the kind of result that our Golden Rule of matching encourages.

But if **SAFE** matches are so difficult to find, why bother creating Merge-Purge Setups based on them in the first place? Fair question; the answer has to do with the end of the process. The final Merge-Purge will use a matching expression that is **less than safe**; for the dupes found by that M-P, we will have to actually examine the Primer Table dupe groups that GoldBox found. It will save us work if we can eliminate all the "easy" matches before we run the final M-P setup.



So here is a **Batch Mode Merge-Purge setup** with the **SAFE** matching mentioned above. Because the screenshot hides some of the Dupe Expression and the SQL statement, that information is shown on the next page.



DUPE EXPRESSION: SmoothComp(Tgt->Company, 'C') + SmoothName(Tgt->Contact) + PhoneStrip("Tgt->Phone1")+Tgt->Title

SmoothComp is a function that does three things:

1. When found in the Company field, the following punctuation is ignored during evaluations for matching: periods, commas, hyphens, parentheses, forward slashes, colons and semicolons.
2. All remaining data in the Company field is **virtually** UPPER cased (does not affect the actual data, but makes the data case-neutral in match evaluations).
3. Every word in the Company field is checked against a custom GoldBox table. If a word is found, it is virtually replaced with its “smoothed” equivalent from the table. For example, “Incorporated” would be replaced with “Inc”. Again, the replacement is virtual, for matching evaluation only; it does NOT affect the actual data.

One of the most interesting things about the SmoothComp function is that it can be applied to other fields. Note that when we applied it above, we included the parameter ‘C’. This causes GoldBox to look only at a “sub-table” made up of replacements we’ve defined for the Company field. (This sub-table is actually shipped with GoldBox.)

But we could also use another sub-table (also shipped with GoldBox) named ‘A’ that’s populated with replacements that are designed for the Address fields. If we then call SmoothComp in our matching expression using an ‘A’ instead of a ‘C’; and apply it to the Address1 field, instead of Company; then it will do the same 3 things noted above, but applied to the Address1 field. In fact, we can create our own sub-tables with names up to 5 characters long (say, ‘TITLE’) and apply the same capabilities to **ANY** field.

SmoothName is another table-based function that looks at the first word (only) of the Contact field and virtually replaces it with the “smoothed” equivalent. For example, “Robert” would be replaced with “Bob”. Again, does not affect the data, just matching.

PhoneStrip is a function that virtually strips anything that is not a numeral from the designated field. This eliminates any matching failures that could arise from formatting variations in phone fields. Again, does not affect actual data, only matching.

SQL statement: CONTACT1.U_COMPANY > '' AND CONTACT1.U_CONTACT > '' AND CONTACT1.PHONE1 > '' AND CONTACT1.TITLE > ''

The SQL statement (or filter) ensures that none of the fields in the Matching Expression are empty. The U_ fields are used where available, because they speed up the process.

In the lower right corner of the setup form, you’ll see the check-marked box for “Copy in Main Contact ADDRESS”. This is just one of several options relating to the **Additional Contact records that are created from Main Contact records that are deleted by the Merge-Purge process**. Creating these Additional Contacts is automatic and not optional. Main Contact data that would otherwise be lost is copied to their Notes.

Also note the option Field DIFFERENCES into the Survivor’s History This is probably overkill, since the Additional Contact will get the same info; but it’s there if you want it.

Another option is the selection of which GoldMine Index to use:  The Index chosen will, when combined with the option NEWEST Contact as Survivor (Default: Oldest) determine which duplicate record will be selected as Survivor. The usual selections are as shown above, which results in the oldest dupe becoming the Survivor.

BACKFILL	Table	Field Name	Field Description	Type	Len	Dcml
2	TGT	COMPANY	Location	Char	40	
2	TGT	CONTACT	Contact	Char	40	
2	TGT	DEAR	Dear	Char	20	
2	TGT	LASTNAME	Last	Char	15	
2	TGT	TITLE	Title	Char	35	
2	TGT	PHONE1	Phone1	Char	25	
2	TGT	EXT1	Ext	Char	6	
2	TGT	PHONE2	Phone2	Char	25	
2	TGT	EXT2	Ext	Char	6	
2	TGT	PHONE3	Phone3	Char	25	
2	TGT	EXT4	Ext	Char	6	
2	TGT	FAX	FAX	Char	25	
2	TGT	EXT3	Ext	Char	6	
2	TGT	ADDRESS1	Address	Char	40	
2	TGT	ADDRESS2	Address2	Char	40	
2	TGT	ADDRESS3	Address3	Char	40	
2	TGT	CITY	City	Char	30	
2	TGT	STATE	State	Char	20	
2	TGT	ZIP	Zip	Char	10	
2	TGT	COUNTRY	Country	Char	20	
0	TGT	DEPARTMENT	Dept	Char	35	
0	TGT	SOURCE	Source	Char	20	
0	TGT	KEY1	Contact Type	Char	20	

Finally, note the **Backfill Survivor Fields** checkbox.

It is accessed via the button, only when Batch Mode is checked.

This controls whether/ how field data from a deleted record is copied into the same field of the Survivor. The codes are shown across the top.

Note that I normally use **2 (NEVER backfill)** for the first 20 fields (names, title, address and phone), and **0** for the others. The reason is that it's not a good idea to backfill partial addresses or phone/ext into the Survivor record. In the case of relocation by a Contact who had only partial information, doing so could give you mixes of old and new info in the Survivor. GoldMine's M/P just backfills into empty fields automatically, unfortunately. With GoldBox, you can make your own Backfill decisions about each and every field.



Here is an example of a Merge-Purge set up for **less than SAFE matching** (and therefore with the Primer Table in mind). Note that the Mode is **Count w/ Preview**; and that Backfill is not checked (it cannot be except in Batch Mode). Otherwise, there is no real difference between this and our first setup (except for the DUPE EXPRESSION and SQL statement, of course). For details on how to "work" the Primer Table, see the article that follows the checklist.

Checklist for Merge-Purge


- Run a procedure to remove all C2 Dupes, if any found by Contact File VALIDATOR (Dedupe C2 Table in GoldBox).
- Run a procedure to remove all C2 orphans, if found (Global Delete of C2s in GoldBox). This is critical; if you do not do it, major data loss may result!!!**
- Run a Filled Field Count report on GoldMine. Consider possible elimination of unpopulated fields.
- Develop as many **SAFE** matching schemes as is practical, considering Filled Fields.
- Create a Merge-Purge Setup with **less than SAFE** matching. Configure all items (except Backfill). Save as **Count Mode, with Preview**.
- Copy (Save As)** above Setup, once for each **SAFE** matching scheme. Change the matching expression in each to one of the **SAFE** matching schemes. Change the copies (but not the original) to Batch Mode. For one copy, configure Backfill; this configuration will be used by all Batch Mode Setups. Be sure to check the Backfill option for all Batch Mode setups.
- The remaining work must be done when Users are out of GoldMine.**
- Optional, but strongly recommended:** One by one, change each Batch Mode setup to Count Mode w/Preview, and run. Open the M-P Primer table and manually verify that **ALL** matches found are true, correct matches. If even one error is found, eliminate that matching scheme from further consideration, and delete the setup that bears it.
- BACKUP GOLDMINE!!!**
- As setups pass the above test, return them to Batch Mode and run the setups.
- Finally, it's time to run the first setup created, the one with less than safe matching. When the run is complete, "work" the M-P Primer Table. Once the Primer Table has been completely worked, change the Merge-Purge that produced it to Batch Mode (**with the special functions for Filter and 1st Dupe in place**) and run it. **IF ANY WORK HAS BEEN DONE IN GOLDMINE SINCE THE PREVIOUS BACKUP, BACKUP AGAIN BEFORE PERFORMING THIS ITEM.**

The following is something I'd suggest for only the most experienced GoldBox Users. The reason is that, for safety, a Q-file Event that will return the final setup to Count Mode w/Preview at the beginning of the next Q-file run is required. Creating that Event is beyond the scope of this document. So, use this suggestion with extreme caution! The following activity would occur AFTER the first warning to BACKUP, above.

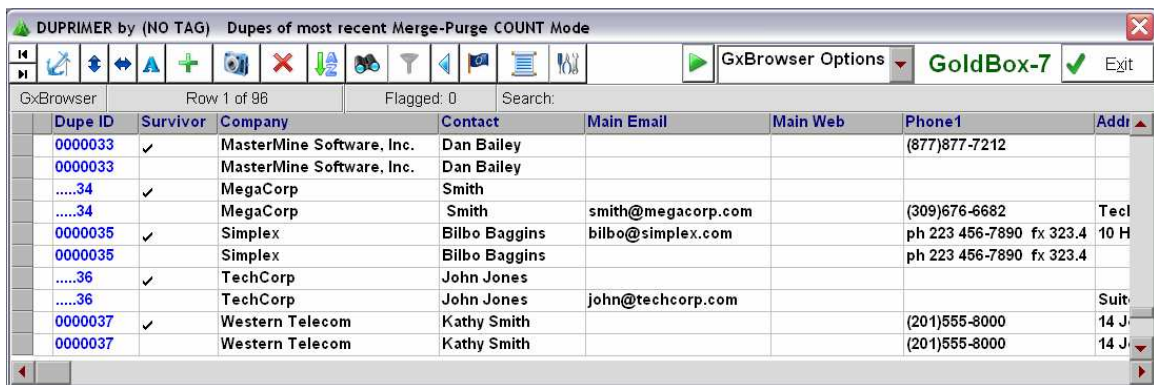
- Create a Q-file, and add all M-Ps to it, with the Count Mode last.
- BACKUP GOLDMINE!!!**
- Run the Q-file.
- Work the Primer Table
- Change the last setup to Batch Mode and run it.
- Change the Mode of the last setup back to Count w/Preview

Using GoldBox's Merge-Purge Primer Table

The purpose of the Primer Table is to allow you to influence what will happen when a Merge-Purge is run in Batch mode (i.e., "for keeps"). **Typically, you will use the Primer Table when running a Merge-Purge that has been set up with matching that is less than SAFE.** The Primer Table is created when a Merge-Purge is run in **Count Mode**, with the **Preview option checked**. Once that "dry run" has been completed, access the Primer Table as shown below:



The screenshot shows the GoldBox-7 2010-A application window. The menu bar includes File, Views, Automate, Conversion, Query, Utilities, Log, Service, Data, Help, and Exit. The 'Data' menu is open, showing options: Open ANY dBase Table, Merge-Purge PRIMER / PREVIEW Table (highlighted), Smoothed COMPANY / GXTranslate() Table, Smoothed FIRST NAMES Table, and SQL Queries / O-FILE Acknowledgments.



The screenshot shows the DUPRIMER by (NO TAG) Dupes of most recent Merge-Purge COUNT Mode window. The window title bar includes 'GxBrowser Options', 'GoldBox-7', and 'Exit'. The table below shows the data:


Dupe ID	Survivor	Company	Contact	Main Email	Main Web	Phone1	Addr
0000033	✓	MasterMine Software, Inc.	Dan Bailey			(877)877-7212	
.....34	✓	MegaCorp	Smith				
.....34		MegaCorp	Smith	smith@megacorp.com		(309)676-6682	Tecl
0000035	✓	Simplex	Bilbo Baggins	bilbo@simplex.com		ph 223 456-7890 fx 323.4	10 H
0000035		Simplex	Bilbo Baggins			ph 223 456-7890 fx 323.4	
.....36	✓	TechCorp	John Jones				
.....36		TechCorp	John Jones	john@techcorp.com			Suit.
0000037	✓	Western Telecom	Kathy Smith			(201)555-8000	14 J
0000037		Western Telecom	Kathy Smith			(201)555-8000	14 J

The numbered groups in column **Dupe ID** each represent a set of "found" dupes. Above, two records per group are shown, but there could be more in any group. The data shown is extracted from GoldMine; although it's possible to edit the entries, you'll accomplish nothing by doing so. If you want to be able to edit data that you see; or just see more data than the Primer Table contains; just **have GoldMine open when you open the Primer Table**. That way, when you click on an entry in the table, **it will be brought forward in GoldMine**, and you can make your edit directly in GoldMine.

There are only 2 types of edits you'll ever make in the Primer Table:

1. Normally, all the records within a **Dupe ID** group will be merged into the survivor record (except the first, which **IS** the survivor record). However, **you can "rescue" any of the records within the Dupe ID group from being merged out of existence by DELETING the record from the Primer Table.**

This is a sort of "double negative" situation; if you delete a record from the Primer Table, that means it drops out of the Merge-Purge Process, so you are deleting it from the list of records to be deleted, which actually saves it from deletion. Of course, if you delete all the records in a Dupe ID group from the Primer Table, the entire Dupe ID group will drop out of the Merge-Purge, and nothing will happen to any of the records. **If you have only 2 records within a Dupe Group; and they are not for the same person; delete BOTH of them. NEVER, EVER leave a Dupe Group with just one (1) record undeleted.** If you do, the M-P process will delete it; odd, but true.

To delete a record from the Primer Table, just press the  button on the toolbar of the Table. If you change your mind about deleting a record, press the button again, and the deletion will be reversed.

2. This type of edit will probably be rare. Normally, the **first Contact record listed within a Dupe ID group will always be the Survivor record**, after the Merge-Purge has been run in Batch Mode. However, **you can change that by placing a checkmark in the column Saved, for the one record that you want to be the survivor record.**
IMPORTANT: NEVER, EVER PLACE MORE THAN ONE CHECKMARK WITHIN ANY DUPE ID GROUP. IF YOU DO, YOU WILL DESTROY DATA!!!

Note: the above 2 items only work because of two special functions that **must** be included in the setup of the Merge-Purge when it's run in Batch Mode. Here's an example:

If you have made any changes using the Merge-Purge Primer Table, the following **MUST** be done before changing a Merge-Purge from Count Mode to Batch Mode. If in doubt, always check with me or another GoldBox expert.

1. You must add the function **InMrgPrimer()** to whatever filter is already in use. If you are using a SQL statement and not a filter, then this function becomes your filter. And...
2. You must use the function **InMrgPrimer(1)** for **1st found Exprn**. Note the use of the parameter 1 in the second expression; it's critical to use this function exactly as shown.

One feature of the Table is that it does contain a numeric summary of the various types of Tab records that are attached to each record (see screenshot on next page). You **MAY** decide to allow this to influence whether or not you change which record will be the survivor. But you would only do that because you feel that the Contact record with the most attached Tab records is likely to have the best address and phone information. You do **NOT** need to worry about losing any of those attached records, no matter which record is the survivor. The survivor will end up with **ALL** the attached records, regardless.

NOTE: new as of November 6: If you prefer a more flexible (and colorful) display of the data in the Primer Table, E-mail a request to me for the GoldBox View of that table (a variation of which is the second screenshot on the next page). It's free; the View displays all fields in the Primer

Table, in the same order. But if you wish, you can suppress any of the fields; rearrange their order; change the labels and colors of the columns; and make other mods. Because the View comes with a (forced) Post operation, closing the View automatically updates the Primer Table itself. I'm available to make mods for you, and show you how to make them yourself, if you like.

Dupe ID	Survivor	Addl Cons	Details	Referrals	Pending	History	Groups	Link Docs	Ap Tracks	Opptyts	Projects	Email Addr
0000033	✓	0	0	0	1	0	1	0	0	0	0	0
0000033		0	0	0	0	0	1	0	0	0	0	0
.....34	✓	0	0	0	0	0	0	0	0	0	0	0
.....34		0	0	0	0	0	1	0	0	0	0	2
0000035	✓	0	0	0	0	0	0	0	0	0	0	1
0000035		0	0	0	0	0	0	0	0	0	0	0
.....36	✓	0	0	0	0	0	0	0	0	0	0	0
.....36		0	0	0	0	0	0	0	0	0	0	1
0000037	✓	0	0	0	5	1	1	1	0	3	1	0
0000037		0	0	0	1	0	1	0	0	0	0	0

Dupe ID (Dupe Group)	Designated Survivor?	Contact	Company	GM Accountno	Primary E-mail
0000034	✓	Smith	MegaCorp	B0080442098\$O.3(L	
0000034		Smith	MegaCorp	A0090540496\$LH8+> Chr	smith@megacorp.c
35	✓	Bilbo Baggins	Simplex	B0042142661\$W2M/L	bilbo@simplex.com
35		Bilbo Baggins	Simplex	B0081626972+4TN3L	
0000036	✓	John Jones	TechCorp	B0080442091%TQ56L	
0000036		John Jones	TechCorp	B0042142662+E/OIL	john@techcorp.com
37	✓	Kathy Smith	Western Telecom	9309130000918000Kar	
37		Kathy Smith	Western Telecom	A6083160868(D9T7LKat	

Finally, if you've ever used GoldMine's **manual** Merge/Purge (because you were afraid to just wind it up and let it run), you know about the time pressure; you can't just lay it aside for an hour or two, then come back to it; you have to cancel and re-run later. With GoldBox's M-P Primer Table, you **CAN** put it aside for as long as you want, and work it at your convenience.

DOCUMENT WRAP-UP

If there's a single idea that I want to make sure you take with you, it is this: for both Imports and Merge-Purge (but especially for Merge-Purge), **MATCHING IS THE KEY**.

There are lots of "score-based" matching systems out there...from the primitive Soundex to GoldMine's points system to sophisticated fuzzy logic systems. But all these systems have the same weakness in common: **they are not binary**. That's important, because whether or not two records are for the same person/entity **IS** binary. It is true or false, it is logical, it is **not quantitative**. If we merge the records of two different people because our points requirement was met, that doesn't help us at all; it's still a catastrophic failure.

Instead of risking our success on algorithms developed by others, based on assumptions that may be irrelevant to us, employing guesswork scoring criteria, GoldBox gives us real control. We decide what can safely be virtually extracted from a field's value, or virtually replaced with a simplified ("smoothed") value. Then, when all these safe, virtual modifications are made, GoldBox uses **EXACT MATCH** as the only acceptable criteria. Then it offers us a manual override. It's the right way to do it.

GoldBox is simply the smart, safe way to eliminate dupes from your database!

Bob Taylor
(904)646-9861
goldboxbob@goldboxbob.com