DataDict -- QUICK INSTRUCTIONS:

1. Give your database a name. Go to menu item Tools/Startup. Enter the name in the Application Title field. This name will be used in the DataDict reports, and is also displayed as the title of the application button on your taskbar.

2. Click the Refresh button; a message will tell you DataDict is working, and another will tell you when it's done. Be patient, please.

3. Click the buttons to see the Data Dictionary and Loading reports on your database. Use the 'Browser' form to see individual tables and their fields.

==========

DataDict

Email: DataDict@email.com
Website: www.CleanDataSystems.com

PURPOSE:

DataDict is a tool to help with documenting a database. It produces reasonable Data Dictionary (Table/Field) and Loading Order reports, and has a 'browser' form that lets you examine the Table/Field information. Its main purpose is to allow you to quickly summarize the contents of your database, and also to make sure that each table and field is fully described and has the property settings you want. It is also very useful for evaluating a database that is given to you without any accompanying documentation.

DataDict is a work in progress. You can help make it better by telling me what you like and don't like, and suggesting additional capabilities.

LICENSE:

The comments here are not a substitute for the full License agreement. The complete License agreement can be viewed from the 'About' button of DataDict, and accompanied the program as a text file in the original distribution archive. Your installation or use of the DataDict constitutes your acceptance of the terms of the License. You assume all risks associated with the use or misuse of DataDict on your system, files, and data.

DataDict is provided to you as freeware. DataDict and its documentation are copyrighted. You are not charged for your copy, can't charge anyone else for a copy, and no registration of any kind is required. Feel free to share DataDict with others, but only in the original archive (zip) format. If you want to thank me: (1) send me an email; (2) visit the website to make a donation; or (3) think of me when you do your next good deed.

REQUIREMENTS:

Depending upon the version of DataDict you are using, you need Access 97, Access 2000, or Access 2002 (XP). You also need at least one defined table with one field in your database.

SUPPORT:

Because DataDict is provided to you without charge, the License should not imply and does not entitle any user to support for solving data problems using DataDict, and it does not guarantee that I will quickly repair any bugs you may discover. However, I use DataDict myself and want to fix any errors that you discover that are not known limitations described below. You can contact me via email at:

DataDict@email.com

I would also like to hear from anyone who finds DataDict useful. Let me know how things should act if you wished they acted differently. Since I only work on this in my mythical spare time, a little encouragement wouldn't hurt either.

DISCLAIMER:

*CAUTION: Always WORK ON A *NEW* COPY of your data, NOT YOUR *ONLY* COPY of your data. You should ALWAYS BACKUP YOUR DATA REGULARLY.

*DataDict is not warranted to be free of bugs. In fact, there is no warranty of any kind. Unlike the incredible diversity of invertebrates in nature, any 'bugs' in DataDict are not to be admired. I've tested and used DataDict extensively, but can't anticipate all of the situations you may encounter with your own data, software and hardware. See the SUPPORT section above to report problems.

USING DataDict

DataDict is simple to use, and was designed to be that way. It takes only 1-3 minutes to load your database information into DataDict, depending upon its size and complexity. (If processing many linked tables it can take much longer--see Known Issues below)

Each time you start DataDict it will display the last database loaded, and when it was loaded. You can preview or browse that information--even if it was loaded from a different database. To freshen the database with the current database's information, just click the button!

1. Click the 'Refresh' button to load the currently opened database's information into DataDict. Use the checkbox to include (checked) linked tables in the dictionary generation (takes longer with linked tables). A message appears telling you DataDict is working. Please be patient while it is doing its job--it may take several minutes. A message will appear to tell you it's done.

2. Click the 'Preview/Print Data Dictionary' button to see a report on the loaded DataDict information. You can print it, save it as an Access Snapshot, as an RTF file (although it will lose some formatting), or send it to Excel for an 'outlined' sheet.

3. Click the 'Preview/Print Table Loading Order' button to see a report on the loading order, parent and child table information for each table in the database. You can print it, save it as an Access Snapshot, as an RTF file (although it will lose some formatting), or send it to Excel.

4. Click the 'Browse' button to see your table information one at a time.

CONSIDERATIONS:

*** On Primary Keys, Relationships, and Foreign Keys ***

* The following comments describe DataDict's detection and reporting of relationship-related properties only. Please see the Access help on "Relationships", "Primary", "Foreign", and "Referential Integrity" for more information on these topics.

*You create defined relationships in the Relationship window of Access. How DataDict reports Foreign Key settings for fields depends upon how you define a relationship, and that is dependent upon whether or not the parent table in the relationship has a unique index on the parent field.

* You can create a unique index on one or more fields in a table; the values represented by the unique index cannot be duplicated within a table, although Nulls in the field(s) can be ignored in determining the unique values. A Primary Key (PK) is a special type of unique index, and a PK may not include Nulls; each record in a table must have a complete, unique PK value. Each of your tables *should* have a defined Primary Key, but sometimes it is not practical. The absence of a unique index (or PK) in a table, however, affects the type of relationship and protection settings you can create.

* When a table having a PK index is joined in a defined relationship to a child table, AND the setting for Referential Integrity (RI) is checked, the related field(s) in the child table are designated as a Foreign Key (FK). Access automatically provides a FK index in the child table for you when you create such a relationship. The PK status of the parent field(s) and the FK status of the child field(s) are reported in the main DataDict report. You will not, however, see the FK index listed in the Index window of the design view for a table; it is not available to viewing or editing because Access considers the FK index as a property of a relationship, and not a property of a table.

* When a table having a PK index is joined in a defined relationship to a child table, but the RI setting is NOT turned on, Access does not create the FK index, and no child table fields will be indicated in the DataDict main report as being Foreign Key fields. RI must be turned on for Access to create the FK index and for DataDict to detect and report the fields involved in it.

* In a relationship where the parent table does not have a unique index, Access will not create a FK index. Even if the parent table does have a unique index, if it is not a PK index, Access will not create a FK index for the child field. The parent table must have a PK index for a child table to have a FK index. Child fields in these kinds of relationships will never indicate a FK status: no PK, no FK.

* Access allows you to set Referential Integrity constraints on a relationship ONLY where the parent table has a unique index (or PK). You may not invoke the Referential Integrity settings for a relationship without a parent table unique index. The implication for DataDict reporting is that the relationship is detected and indicated in the Loading Order report, but no FK setting will be seen for the child table field(s) in the main Table/Field Report without a PK parent in the relationship.

* Linked tables don't show their relationships automatically in the Relationship window, although foreign key (FK) fields are detected by their FK index. Integrity constraints are enforced in the tables themselves, not in their links. By joining linked tables in the Relationship window of Access you can adjust the relationship settings in the current

database, but not at their source (there is no relationship in the *current* database by default). Yeah, this is confusing stuff. It's best to work with linked tables where they actually reside. Read the next section.

*** Table Dependencies

Both the main data dictionary report and the loading order report show a field called Dependency. This field indicates the extent to which a table is dependent upon values from another table through a foreign key (FK) field. Recursive (self-join) FKs are not considered. The following five Dependency conditions are identified:

** Independent - table contains no FKs from other tables; tables with simple joins without referential integrity turned on may be classified as Independent despite implied dependencies

** Full - table is fully dependent upon another table because it contains a FK as part of its primary key (PK)

** Partial - table is partially dependent upon another table by having at least one required, non-PK, FK field

** Optional - table is optionally dependent upon another table by having at least one non-PK FK field that is not required

** Unknown - tables that are broken links or have other connection defects have unknown dependency.

*** Linked Tables and ODBC ***

* Linked tables are processed much slower than local tables. It is best to use DataDict within the database file where the tables actually reside, rather than from a front-end. If you do run DataDict against a database containing linked tables, check the option above the Refresh button and then please allow extra time for DataDict to do its job.

* Linked tables don't show their relationships automatically, although foreign key (FK) fields are detected. That's because there is no relationship in the *current* database.

* Broken links are detected and reported as such.

* A future version of DataDict will report the source type of ODBC-linked tables. In this version, the information provided about an ODBC link is incomplete. See the Linked Table Manager applet to manipulate your links and see their properties.

*** Other ***

* When viewing the DataDict information about your database, you may want to edit it. DataDict is reporting FROM your database and not just displaying information you've typed into an unrelated report. Fix whatever is wrong in your database and then refresh the dictionary information.

* DataDict can grow in size over time because it stores your database information in its own tables. Each time you refresh the dictionary information there is more junk space left in the DataDict.mda file. The solution would be to compact DataDict, but it can't be done in code (= I can't make a button to do it for you). But, you can overwrite the installed copy of the Add-In with a fresh copy. Or even better, check the www.CleanDataSystems.com website to see if there is an update and install that!

KNOWN ISSUES:

* Please see the text above on the detection and reporting of Foreign Key status for fields.

* Please see the text above for information about Linked tables--they are slower to process and may not reveal their relationships, although foreign keys may be detected. It is best to use DataDict where tables reside, rather then detecting them through links. You must use the checkbox option to include linked tables in the dictionary generation.

* One site where DataDict was used caused Access itself to lock up when DataDict was being refreshed. The conditions were rather unique. The tables causing the problem were links to another database across a network. The work site also had license management software running in front of Access; users must log in each time they open a fresh copy of Access on an mdb file. I believe the combination of security on Access (and perhaps the back-end database?) and that this was a link across a network caused the problem. Access stalled, and the Task Manager reported it was "no longer responding". When that instance of Access was shut down in Task Manager, the .ldb file left behind had to be deleted before the database could be opened for editing again (See Access Help on the ldb, or locking file). No other occurrence like this has been reported. The solution was to copy the troublesome tables to the local mdb file, and then run DataDict.

TIPS:

* Give your database a title. DataDict will use the title in its reports. The title will also appear on the taskbar button when the database is open (handy if you have more than one open at a time). Go to menu item Tools/Startup and enter an appropriate name in the Application Title slot.

* Provide all of your tables that are involved in relationships with a Primary Key. This will automatically create a FK reference in the child table(s), and allow you to set

Referential Integrity constraints on the relationship. These are *good* things, and you should learn how to use them if you don't already.

*** MAKE YOUR DATABASE SELF-DOCUMENTING ***

If you use the description areas for Tables, Fields, Queries, Reports, Forms, and Modules, you will have done a lot to make your database understandable to others--or even yourself in the future. These descriptions stay in the database and travel with it everywhere it goes. You can't lose them unless you manually delete them! You can view the descriptions at any time using the menu item View/Details.

* Give your Tables good descriptions. It is very handy to be able to *read* someone's comment about the function or contents of each table, instead of trying to decipher cryptic table names having even more unintelligible field names inside. In the database window, Right-Click on a table and select Properties. There is a window waiting for the description. DataDict will display the description in its Data Dictionary report.

* Give your Fields good descriptions. We are all busy, but you should take the time to write good descriptions. You only have to do it once. Open a table in design mode, and type the description in the space provided next to each field. This description will be shown in the Status area window (lower left part of Access window) whenever you are in a field in an open table, in a form, and in many queries (where the field has not been manipulated by functions or renaming). Include any information the user needs to know to understand the data. For example, if numeric values are shown in a field, give the measurement units in the description. DataDict will display the description in its main Table/Field report.

* While you're at it, give your Queries good descriptions, too. You can make shorter and more logical query names if you use the description space for an explanation of the query function, or its position in a series of queries. Descriptions can also be written for forms and reports. Try it.

Explore and have fun. I hope DataDict helps you to make your databases more accessible to others, and also gives you some insight into poorly documented mdb's that are handed to you. Let me know what you think.