Improving Data Quality with Open Mapping Tools

February 2011

Robert Worden

Open Mapping Software Ltd

Contents

Notes

1. Introduction: The Business Problem

Providers of health and social care are under immense pressure to reduce costs, while maintaining or improving outcomes. They must find practical measures which will improve their costs and outcomes in the short term, while laying the foundations for sustained improvement in the long term.

In this context, attention can usefully be focused on the costs of poor data quality, and the opportunity to reduce those costs.

Providers rely on information stored in a range of databases and documents throughout their organisations. For a host of reasons, this information becomes incomplete, inaccurate and inconsistent. To the same extent, the business processes and care processes which depend on it become flawed and inefficient. The causes and manifestations of the problems are diverse – but the end result is that poor data quality costs lives and blights lives.

In the past, insufficient attention has been paid to data quality, for several reasons:

- There is insufficient visibility of the extent of data quality problems
- There has therefore been no means of assessing their knock-on costs to the business, or their effects on quality of service and outcomes.
- Even when the scale of the problems is appreciated, the technical tools to tackle them are specialised and expensive – beyond the budgets of many provider organisations
- Those IT suppliers who provide systems for providers do not have specialist data quality tools to offer them.

Open Mapping Software now offers a powerful integrated toolkit for tackling data quality problems, at an acceptable cost. The full lifecycle of data quality improvement – from initial exploration, through quantifying the problems and consequences, bulk detection of defective data, and bulk correction of data - can be addressed in an integrated manner with a single toolkit. Effective data quality improvement initiatives can be cost-justified, planned and executed, on any scale from local to enterprise-wide.

The toolkit may be used in standalone form by providers of health and social care, or it may be more closely integrated with the product sets of their specialist IT suppliers.

Notes

This white paper describes how the Open Mapping tools can be applied through the full lifecycle of a data quality improvement programme.

2. Initial Assessment: Understanding the Data Quality Problem

Many organisations do not even know where their data quality problems lie, let alone appreciate their extent and their impact on performance. The first step is a broad initial assessment of the nature and location of data quality problems.

A key part of this initial assessment involves 'MBWA' or 'Management by Walking About' – simply going and talking to the people who use the organisation's information at the front line, and find out where they experience problems from poor information quality. This is likely to lead to a short-list (or long list) of key suspects – information sources whose inaccuracy, inconsistency, or incompleteness degrades the performance of the organisation.

Having identified the key suspects – the document collections, databases, sets of spreadsheets or other information whose quality undermines performance – you need to assess the nature and extent of quality problems in those information stores; and here a technical problem arises. The many different physical forms of the data stores make it hard to know what they each contain, and even harder to compare their different 'versions of the world' side by side.

Typically an organisation will have dozens of even hundreds of different databases, each with its own complex database schema. The diverse schemas present a technical barrier to knowing what is in the databases, and to comparing them side by side for data consistency.

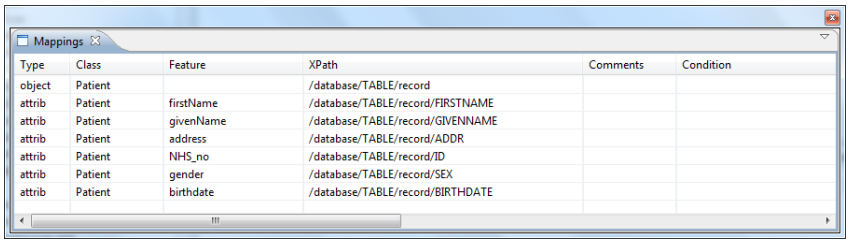
The Open Mapping tools provide a rapid and cost-effective way to overcome this barrier, by mapping all databases (or spreadsheets, or XML document collections) onto a common logical information model. Then the information in all the different databases can be viewed and explored side by side, in terms of the logical model, without needing to understand the detailed database schemas of each database.

The initial 'MBWA' assessment will have identified a few key types of information whose quality is suspect and undermines performance. The initial logical model need only cover these types of information, and so need not be complex. It may, for instance, cover only basic person demographic data, in order to tie together information about the same person held in different data stores.

Notes

As the initial logical information model is simple, so the process of mapping each database schema onto that model is also simple, and is supported by the tools. The mappings define how each database represents the logical information in the model. They can be captured and validated with a dedicated **Mapping Editor**, reviewed in a tabular spreadsheet form, and further validated by running queries against the database. This requires technical understanding of each database schema, to be checked with the designers or developers of that database; but once this knowledge is captured in mappings, specialist knowledge of the database schemas is not required for any later steps of the investigation.

The screenshot below shows, in tabular form, some of the mappings of a simplified patient database onto a logical model of patient demographic data.


Mappings

Type	Class	Feature	XPath	Comments	Condition
object	Patient		/database/TABLE/record		
attrib	Patient	firstName	/database/TABLE/record/FIRSTNAME		
attrib	Patient	givenName	/database/TABLE/record/GIVENNAME		
attrib	Patient	address	/database/TABLE/record/ADDR		
attrib	Patient	NHS_no	/database/TABLE/record/ID		
attrib	Patient	gender	/database/TABLE/record/SEX		
attrib	Patient	birthdate	/database/TABLE/record/BIRTHDATE		

(In this simplified example, there is only one table in the database, and its name is 'TABLE'. More realistic examples will be provided soon.) The table of mappings says in essence: 'this field in the database represents this attribute in the logical model'. Data conversions between database fields and logical model attributes are supported in the mappings.

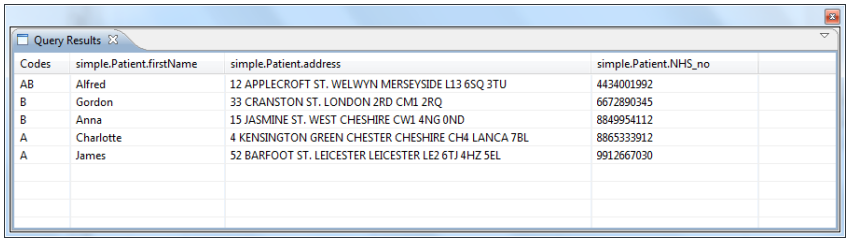
Once you have mapped one or more databases onto the logical model, and connected the Open Mapping tools to the mapped databases (typically by odbc) you can use the mappings in a **Comparative Query Tool** to explore the information in one or more databases, individually or side by side. To do this, you write queries in a query language which depends only on the logical information model – with no mention of database tables or fields. Users can write queries with no detailed knowledge of the databases. The text of a typical simple query is shown here:

```
select  Patient.firstName Patient.address Patient.NHS_no
```

The query uses class names and attribute names from the logical model. Useful queries can be as simple as this, or may involve conditions on any attributes, or navigate any associations between classes of the logical model.

The tools use the mappings to convert the query into SQL against any mapped database to run efficiently. The query tool runs the query simultaneously against one or more databases, collects the results from all databases, and displays them together in a single table. Results are presented entirely in terms of the logical model, not the database fields. So users can understand query results without any knowledge of the databases. The result of the simple query above, run against two databases (denoted by codes A and B in the result table) is shown here:

Notes



Codes	simple.Patient.firstName	simple.Patient.address	simple.Patient.NHS_no
AB	Alfred	12 APPLECROFT ST. WELWYN MERSEYSIDE L13 6SQ 3TU	4434001992
B	Gordon	33 CRANSTON ST. LONDON 2RD CM1 2RQ	6672890345
B	Anna	15 JASMINE ST. WEST CHESHIRE CW1 4NG 0ND	8849954112
A	Charlotte	4 KENSINGTON GREEN CHESTER CHESHIRE CH4 LANCA 7BL	8865333912
A	James	52 BARFOOT ST. LEICESTER LEICESTER LE2 6TJ 4HZ 5EL	9912667030

(The example uses two local databases with very small numbers of records; but queries run efficiently against databases of any size, using database indices automatically where they exist).

These results show the records retrieved only from database A; the records retrieved only from database B; and records denoted by the code 'AB' where identical information was retrieved from both databases. Query results can be instantly sorted on any column. The table shows that the two databases agreed on all three attributes for only one patient. Sorting the records on different columns can reveal 'near misses' for other patients.

The query tool is a rapid and powerful way to explore and compare data across several databases – without detailed knowledge of their schemas (that knowledge has been encapsulated in the mappings). Users can run queries, browse the results, refine the queries, bring in other databases, and store query results, for instance in spreadsheets. Doing this can expose and characterise numerous data quality issues.

For instance, you can write a trivial query which returns only the NHS numbers of patients. Run this query against two databases A and B. Sort the results by database code and find the records which have code 'A' (not 'AB'). These are precisely the patients that are known in database A, but not in database B. You have identified the missing patients in database B, with one simple query.

3. Bulk Detection of Data Quality Problems

Not all data quality issues can be detected by simple queries, often because of small errors and inconsistencies in the data. In a large data set, there will be significant numbers of records with small errors in various fields; with spelling errors and omitted words in names and addresses; and with near-miss duplicate records in the same table. Exact matching of records and attributes, as done by the query tool, does not catch these near-misses.

In these cases, the computational challenge is to detect duplicate records and matching records by approximate matching – which detects the near misses as well as it detects the exact matches. This is a hard problem, and standard database tools do not do it well. Doing it by exhaustive matching of all record pairs is an N^2 problem; the number of record comparisons grows as the product of the sizes of two tables, or as the square of one table size. Even with 30,000 records – by no means an exceptional database size – such a brute-force match requires a billion record comparisons, so the search cannot be done in rapid exploratory mode – or even done at all, in some cases.

Notes

To meet this challenge, specialised data matching software has been developed. However, it is usually only available from dedicated data quality tool suppliers such as FirstLogic and Trillium, and is sold to large commercial organisations at a very high price tag. These tools are usually outside the budgets of providers of health and social care, and are not in the software offered by specialist suppliers in their sector. So bulk data matching and de-duplication does not get done, and the quality problems persist.

The Open Mapping tools include a capability for fast data matching and de-duplication, which finds approximate matches between records very quickly – between 10 and 100 times faster than by exhaustive record matching, depending on the matching criteria. This performance is competitive with that of the most expensive high-end matching packages. More important, it enables the rapid iterative investigation of data quality problems in large data sets.

It is usually not possible to find the right criteria for approximate record matching at a first pass, so as to minimise the number of false positives and false negatives. Doing so requires rapid experimentation and feedback to tune the matching criteria. This is not possible if each test run takes many minutes or hours. The fast matching of the Open Mapping tools makes it possible.

Two distinct fast matching methods are built into the tools, and can be combined in a single match:

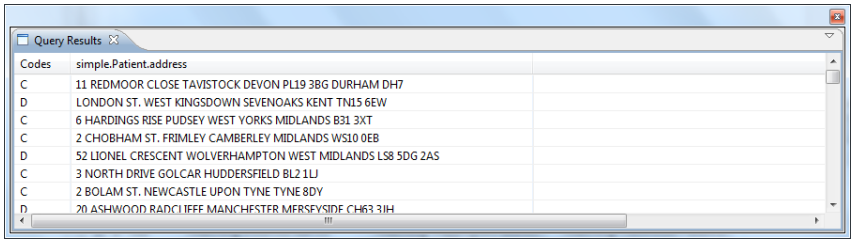
- Matching of fields in the presence of a defined maximum number of single-character errors (e.g. NHS numbers with up to three digits incorrect); so-called 'fuzzy matching'
- Bayesian maximum likelihood matching of fields such as names and addresses, where words may be reordered or missing, as well as having individual spelling errors and variants.

Sometimes a matching problem can be tackled with one of these methods, sometimes the other, or sometimes both in combination are needed. The Open Mapping tools include a **Matching Workbench** where these matching criteria can be tried out and combined, in rapid experiments to find the best matching criteria.

The input to the matching workbench is a result set from a comparative query, as described above – one or two large tables of data where the column names and data content are defined in terms of the logical model, without reference to database tables and columns. The user of the matching workbench does not need to know the detailed databases structures, which have been captured once in the mappings.

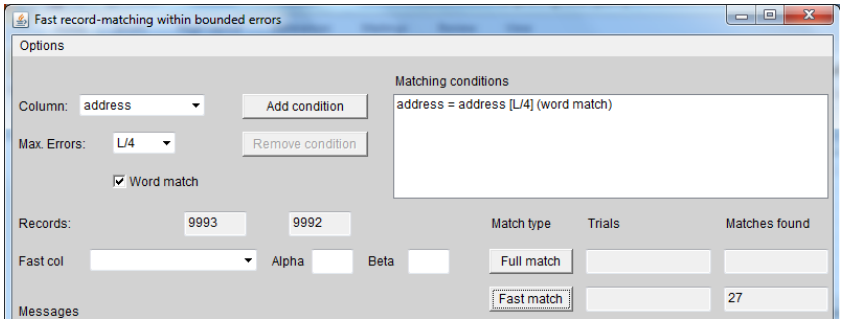
Any query can be run in fast matching mode, against one database (for detection of duplicate records in that database) or two (to detect matching records in the two databases). The query may return many thousands of records from each database for matching.

First, a sample of up to 50 records from each data source are shown in the query result table. For instance, when matching addresses from two databases each with 10,000 patient records, this might look like:



This first view simply gives the user a chance to see the general nature of the data in each data source, in order to frame initial matching criteria. Typically the user will retrieve data for more than one attribute of the logical model, to give a choice of matching fields.

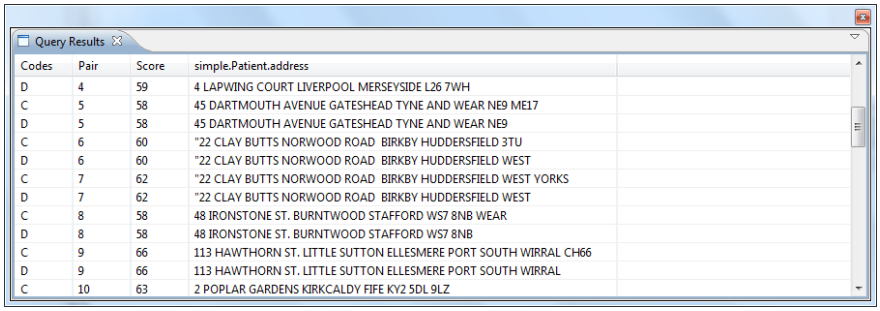
At the same time, the matching workbench window is displayed:



This gives controls for the user to define one or more matching conditions, and then run a fast match. In this case, the query has retrieved 10,000 address records each from two databases; the user has set up a single condition for word matching between the addresses in each database; and has run a fast match, finding 27 approximate matches. The run took about 5 seconds, so the user can easily tune the match parameters and run it again.

The workbench window gives settable parameters for the logical matching conditions, as well as physical tuning parameters which are sometimes needed for best performance on very large data sets. It is generally not necessary to vary the physical tuning parameters.

At any time, the user can go to the query result table and display the matching pairs of records from the latest match:



These matches do not show many spelling errors or re-ordering of words, but sometimes show some strange additions to the postcodes in either address. From seeing the characteristics of the matches, it is possible to refine the criteria for the next match.

Notes

When this exploration has reached a useful set of matches, the matching records can be exported to Excel for further investigation. Only rarely do the matching criteria result in a data set that can be used immediately to correct data in one or other database; usually it will be necessary to apply further manual review before applying data corrections.

4. Assessing the Business and Human Impact of Data Quality

Carrying out an exploration and detailed analysis of data quality issues can lead to some unpleasant surprises. It may be that the level of duplicate records in some databases, or the level of missing or inaccurate records in others, turn out to be much higher than was previously thought from anecdotal evidence.

If so, this may not be all bad news. Combining the qualitative evidence from interviews with front-line staff, with the quantitative data on the extent of the root problems in the data, may lead to a better quantitative understanding of the costs of the problems – both the direct costs to the organisation, and the human costs to those it serves.

This in turn may lead to a sound business case to invest in addressing the data quality problems – with clear quantified expectations for reductions in costs, quicker return on investment, and improvements to service quality. If this leads to positive, cost-effective action, the investigation of data quality will have justified its costs.

Any organisation in the public service has an obligation to ensure its budgets are used most effectively, and are seen to be used most effectively. This is especially true in the current era. It should not be open to the charge either that it failed to understand the hidden costs of poor data quality – when it could easily have done so – or that it failed to act to correct them.

5. Correction of Data Quality Problems

The query tool and data matching tools give a powerful way to detect data quality problems, to home in on defective data and to record it in stored machine-processable forms, such as defect databases and spreadsheets.

Once these records of defective data have been extracted, and subjected to sufficient review to have confidence in the assessments, the next step is to re-apply corrected data to the deficient data sources. For instance, it might be established that patient data in one database is of higher quality than in another database, so that the first can be used as a master database for correction of the second.

It would of course be possible to apply these corrections to the second database manually – but it may be prohibitively slow and expensive to do so. The Open Mapping tools provide some key building blocks for automation of the process.

All the different databases have been mapped to a common information model. Up to this stage, the mappings have been used to re-express logical queries as SQL against the databases, and to transform data 'in' from the forms held in the databases, to a central form depending only on the logical model.

Notes

However, the same mappings can equally be used to transform data 'out' from the structures of the logical model to the data structures used by any database – or to transform between the structures used by the different databases.

Therefore the mappings can be used to transform records arising from the query and matching processes, to a form closely matched to the structure of any database – as input to a simple update program for that database. Or they can be used to transform records extracted from one database into a form matching the structure of another.

Implementing the data migration processes needed to improve data quality will generally require design decisions specific to the problem in hand. But the data transformations derived automatically from the mappings provide an important element of support to this process, by bridging automatically between the different physical data structures of the various databases.

.

6. Sustaining Data Quality

Once a data quality improvement initiative is under way, the organisation will begin to notice and appreciate the effects of the improved data quality. However, this is of little use if data quality is allowed to slip back to its previous levels, by failing to change the underlying processes which create and maintain that data.

Better appreciation of the nature of data quality problems leads to a better understanding of the sources of those problems, which in turn should lead to actions to correct those sources. The Open Mapping Tools can assist in several ways:

- Poor data quality is often a consequence of poor data integration. If the same data are maintained independently in three separate data silos, it will cost more and be done more poorly that if the data are maintained in just one place. By bridging the gaps between different physical data structures, and so allowing easy transfer of data between different systems, the Open Mapping tools make better data integration much more achievable, once the business case for it has been made.
- The same data quality probes which were employed one-off in the initial data quality assessment can also be employed continuously in Business As Usual. For instance, the mappings and comparative query tools can give every data owner the means to continually monitor the quality and consistency of his 'own' data compared with overlapping data in other data stores. Ongoing data quality targets can be set and monitored.
- Data quality checks can be employed at the point of creation of data, in cost-effective ways. If, when a patient record is created, you need to find approximate matches of that record against thousands of records in one or more databases, in order to avoid duplicates, then because of the fast matching capability, such tests are likely to be

feasible to do, within acceptable response times. Without fast matching, they are not.

Notes

Instilling and maintaining a data quality culture will permanently raise the efficiency and effectiveness on the organisation. The Open Mapping tools can help make this process much easier and less costly to do.

7. Supporting the Data Quality Lifecycle

This paper has shown how the Open Mapping toolset can provide integrated support for a complete data quality improvement lifecycle – from initial exploration through bulk detection of defects, through bulk corrections of data to maintenance of improved data quality.

These tools achieve high productivity and rapid results. The approach of mapping to a simple logical information model reduces complexity and learning time for participants, and considerably reduces the time taken in software interface development. The use of a single integrated platform for all stages means that information can pass between the stages without loss or delay.

After a short initial business investigation, it takes only a few days' technical effort to design the logical information model (if there is no pre-existing model, which there usually is), to map some key databases onto it, and to run exploratory queries on those databases. As little as five day's effort can get you to this point. This stage on its own can produce unexpected insights and provide the justification for fuller investigation – which in turn makes the business case for a data quality improvement programme, large or small.

Because of the productivity and effectiveness of the tools, a 'one step at a time' approach can be taken – authorising each new stage only when the previous stages have made the business case for doing so. From a very small initial investment (or even a free proof-of-concept exercise), organisations can embark on a programme of progressive data quality improvement, which can reduce costs and improve effectiveness in diverse ways, unique to the organisation.