Technology Selection for Identification Applications

Ed Sharp
Auto-ID Lab, University of Cambridge, UK

Report Abstract: This report documents the conversion of a theoretical model for selecting the right identification technology for a given application into a working tool that assists the user through the selection process.

Contents

1. Introduction

This report aims to develop a demonstration of the ID application matching tool using first approximation of criteria and considerations.

The following aims were set out at the February 2006 Aero-ID board meeting for the ID application matching track:

- Review the characteristics of different ID technologies

- Examine the complexities of selecting ID technologies

- Propose the next steps towards developing a software application to assist the selection process

This report furthers the work of the ID application matching track outlined above.

From the user's perspective, the tool is split into the following four phases:

- Answering questions about the application and the environment

- Visually refining the values of several criteria to reduce the number of potential technologies to a manageable number

- Using the report generated by the tool to investigate the remaining technologies

- Putting the results of the investigations back into the tool in order to be able to repeat the selection process with the new values, if a technology has not, by that stage, been decided upon

2. ID Technology Selection

This chapter presents the theoretical basis upon which the tool was developed. It has to be noted that this is still work-in-progress and Edouard Thisse's paper, when it is released, will fill in the details.

2.1. The technology selection process

Work done by the Auto-ID Labs has resulted in the identification of six main forms of automated ID technology as listed below. All these identification technologies offer methods of storing some form of identifier in a way which can be read by a machine at a later date.

- Barcodes—includes Direct Part Marking (DPM), 1D and 2D barcodes

- Contact Memory Buttons (CMB)

- Magnetic stripes

- Optical stripes

- Radio Frequency ID (RFID)—includes "Smartcards" with an RF interface

- Smartcards (SC)—restricted to electrical contact interface

In his research[1], Edouard Thisse identified a way of capturing the characteristics and limits of these technologies. In addition, for each of the following four areas, he has identified the criteria and the units that the technology selection process would rely on.

- The concept

- The codage

- The tag

- The reader

To turn the above parameters into a practical application, they were categorized from these four areas into the three levels described below.

- Level 1 – Concept: It encompasses all the characteristics of a technology that will always be true for that technology. For example, a Contact Memory Button will always require contact to be read or written.

- Level 2 – Theory: It encompasses the standards governing the storage of information. For example, the Datamatrix defines the way information can be encoded into the form of a two-dimensional (2D) barcode.

- Level 3 – Practice: It encompasses the specifications of the physical objects—readers, writers and media—in the market. For example, the MacSema Micro-button stores 4KB of information.

For a given application, a company would start by identifying all the available technologies. These can then be eliminated at the conceptual, theoretical and practical stages, based on failure to meet the application's criteria. Until this point, the company would be largely relying on manufacturers' specifications. Once the number of technologies has been narrowed down sufficiently, the company can afford to test the remaining technologies and build a business case if a technology is found to be suitable. The chosen technology would then be introduced, probably as a small-scale pilot initially, followed by company-wide rollout if the pilot is successful.

---

[1] To be announced: paper from the Cambridge Auto-ID Lab

2.2. Assisting the selection process

In general terms, a company can adopt any of the two approaches for selecting ID technologies. The first is to conduct research on a particular technology and develop a knowledge-base as to how the technology can be used and what its limitations are. With this knowledge, the company can search for potential applications for the technology. The tool described in this report will not specifically try to help this approach.

A second approach can be adopted when a company has a particular application in mind and needs to find a technology that can meet the constraints of that application. It is this approach that the ID technology application matching tool will be of assistance.

As soon as a particular application is decided upon, the user will be able to input the constraints of that application and the tool will start to filter out technologies at the conceptual, theoretical and practical levels.

We decided that the best approach to filter technologies is to establish a set of criteria, common to all the technologies, by which each technology could be compared against the others. The process of selecting a technology starts with a complete list of all the potential technologies for which the tool has information. At this point all the technologies are "live". As constraints for each criterion are added, the technologies that no longer match the constraints are eliminated from the list and become "dead".

3. The Tool

This chapter introduces the functionality of the tool; the user interfaces; and the way a user would interact with the tool.

3.1. Function

As outlined above, the main function of the tool will be to filter out technologies based on the criteria input by the users. A secondary function of the tool will be to allow the users to input the results of their own experiments on particular technologies, enhancing the tool's default values and relationships when they use the tool subsequently.

3.2. Interfaces

3.2.1. Application and environment questions

At the time of writing this report, this interface had not yet been realized.

The aim of this interface is to gather as much information as possible about the user's application and environment in order to be able to set the initial conditions for filtering out inappropriate technologies.

The tool takes a series of questions stored in the database and asks the user to answer them. Based on the answers, the tool might ask some follow-up questions. When there are no more questions to be asked, the tool will move on to the criteria filters interface. The process is described in more detail in section 4.1.2.

Questions are limited to the following answer types:

- Yes/No (check box)
- Some numerical value (text box)
- Single selection from a list of several options (drop-down box)
- Multiple selection from a list of several options (list box)

3.2.2. Criteria filters
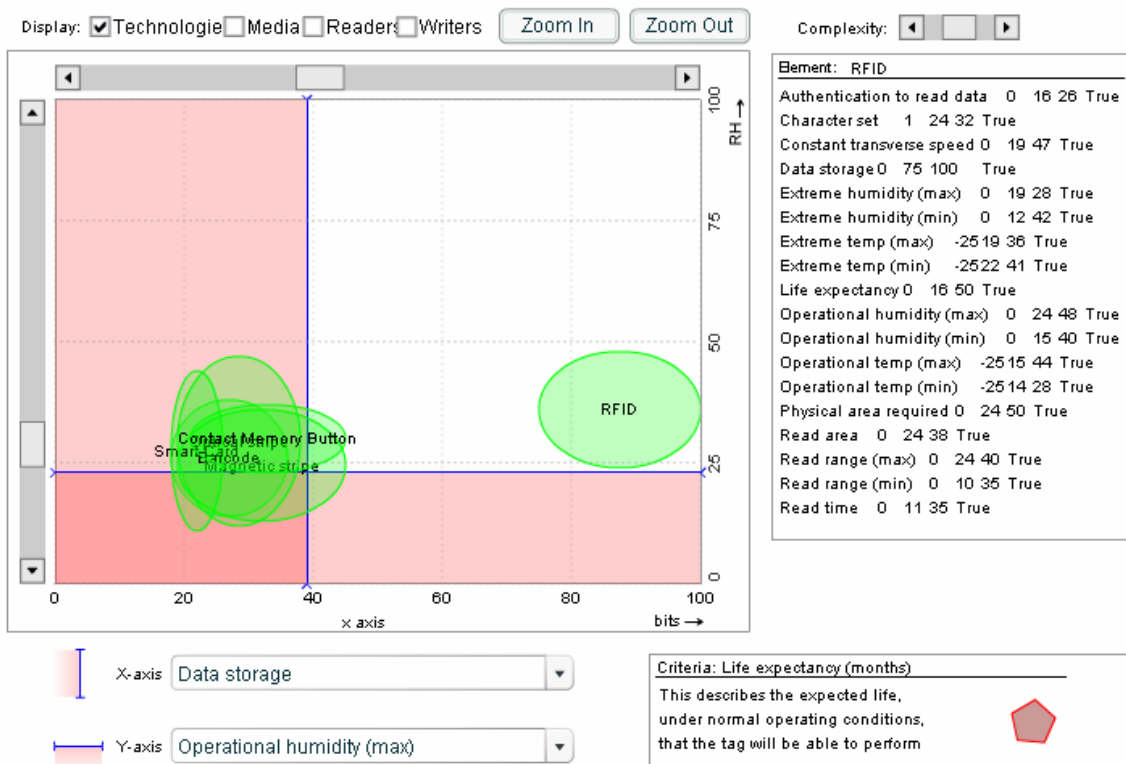
Figure 3.1: Screen shot of the criteria filters interface

The main feature of the interface shown in Figure 3.1 is the 2D space used to represent the technologies' performance on two independent axes. Each axis can be set to display one of the following 14 criteria:

Data storage (bits)	Life expectancy (months)	Operational temperature (ºC)
Read range (mm)	Constant transverse speed (m/s)	Extreme humidity (RH)
Simultaneous reads (number)	Character set (bits/character)	Extreme temperature (ºC)

Read time (seconds)	Authentication to read data (cryptographic level)	Physical area required (mm^2)
Read area (mm^2)	Operational humidity (RH)	

Each criterion is described in more detail in section 5.1.

The value of the filter for each criterion can be changed visually, by dragging the scrollbars at the left and top of the 2D space. The red blanking indicates the technologies that are filtered out by that criterion. There are three types of threshold for the filter:

- More than: The technology must be more than a certain value otherwise it is filtered out. E.g. life expectancy must be more than 12 months.

- Less than: The technology must be less than a certain value otherwise it is filtered out. E.g. read time must be below 0.5 seconds.

- Between: The technology must be between a range of acceptable values. E.g. operational temperature must be between 0ºC and 55ºC.

The technologies that have already been filtered out by the remaining 12 criteria, not on the currently displayed axes, are displayed as red ellipses. The green ellipses are technologies that, provided they are not covered by the red blanking, meet all the criteria imposed so far.

The "level of detail" scrollbar alters how the data is aggregated. At the lowest level of detail, one ellipse represents each of the six technologies. At the greatest level of detail, individual media configurations (explained in section 4.2.1) are displayed.

3.3. Methods of operation

The following sections describe how the tool would be used by different categories of users — the technology manufacturer, the tool administrators and the end-user.

3.3.1. Method of operation – Technology manufacturer

Manufacturers of media, readers, writers and sub-components (e.g. antennae) will be able to input and maintain their products in this database. This information will drive level 3 — the filtering of technologies based on practical criteria.

It will be in the interest of the manufacturer to keep adding their latest products into the database in order that an end-user might choose them. When information is entered by a

manufacturer, it will not go directly into the live system, but will be temporarily held to allow a moderator to check the information.

3.3.2. Method of operation – Administrator

3.3.2.1. Updating information

The administrators have to keep levels 1 and 2, the conceptual and theoretical levels up-to-date. This will involve following all the technology fields and making new entries into the database whenever necessary. Similar to the manufacturers, entries will be temporarily held for a moderator's inspection before being added to the live system.

3.3.2.2. Moderating information

The administrators are also the moderators of the data input by the manufacturers. The moderation process is described in section 4.5.1.

3.3.2.3. General administration

The administrators will be responsible for adding companies into the tool. As will be elaborated in section 4.4.1, there will be many different types of companies stored in the tool. For each company there will be contact information to maintain.

Companies may also have users that have the ability to log on to the tool. These users have information stored such as their usernames and passwords. The administrators will be responsible for maintaining this information, too.

3.3.3. Method of operation – End-user

3.3.3.1. Phase 1: Application questions

In this phase, the user requires to answer questions about the application using the interface described in section 3.2.1. Whilst there is no obligation to answer any questions, each answer will in some way help to filter the technologies.

3.3.3.2. Phase 2: Refining the criteria filters

Having finished answering the application questions, the tool calculates initial conditions for the 14 filter criteria. The tool will switch to the criteria filter interface described in section 3.2.2. Some technologies will immediately appear as red ellipses—the initial conditions will have filtered them out already.

Phase 2 involves tweaking the values for the filters on each of the 14 criteria until the user is left with a smaller set of technologies. As the user reduces the number of technologies, the level of detail displayed can be increased, to be more precise in the filtering.

3.3.3.3. Phase 3: Reporting on technologies

Once the user has reached the desired number of technologies, the tool produces a report. For each technology, the specification is given, links to datasheets may be provided and there may be warnings about factors that might need further investigation. For example, if the user had answered in phase 1 that the product concerned has a lot of metal, then in the report, next to an RFID technology, there may be a warning that metals have an adverse effect on read range and therefore further experimentation is needed.

3.3.3.4. Phase 4: Entering experimental data

Continuing from the example in phase 3, if the user then did an experiment with the aforementioned RFID technology and discovered a value for the read range, the user could enter that value into the system. The user can now continue the iterations in phase 2, but using the new value for read range and therefore allowing the filtering to be more accurate.

For applications where the manufacturers' specifications have to be verified before the part can be used, there will be the facility to validate or override manufacturers' specifications in a similar manner to that of overriding a given value with an experimental value.

4.   Design of the Application

The following chapter presents the design of each section of the tool, indicating which bits of the database and what relationships govern the way that phase works.

4.1. Design of phase 1

The database schema with all the tables is included in Appendix 0.

4.1.1. Database information

Eventually the whole process of asking questions and processing of answers should be totally dynamic and driven from the database. This will allow the administrators to readily modify, add or remove questions from the system.

The database contains the following elements that are directly related to phase 1.

Table "geographies": This is a hierarchy of geographical regions, with the whole world at the top level and smaller regions below it.
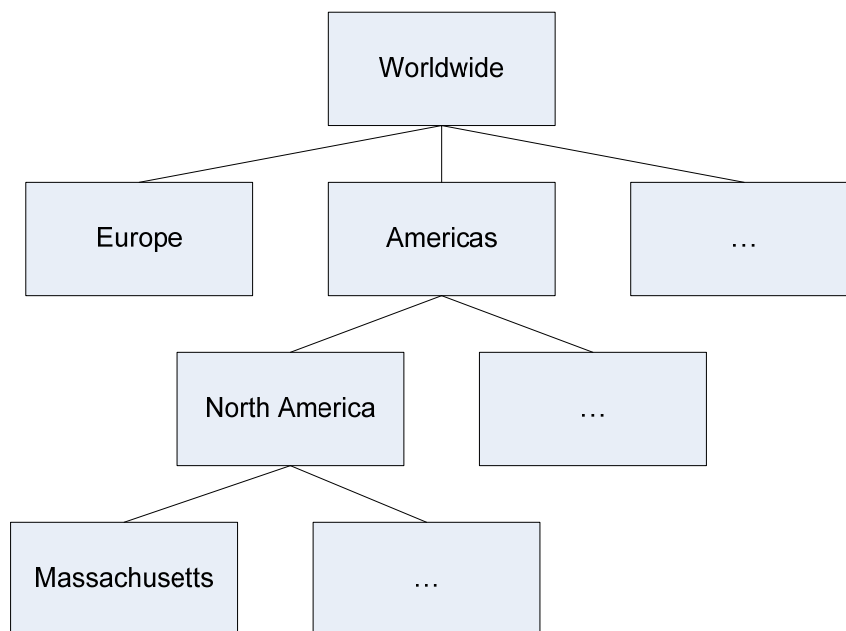
Figure 4.1: Sample hierarchy of geographical regions

There is no limit to the hierarchy, so regions can be added as necessary whenever a new region is needed. Figure 4.1 shows an example hierarchy going down to the state level.

Table "industries": This is a hierarchy of industries in which each industry is linked to the geographical region in which it operates. E.g. the British postal service is linked to the United Kingdom.
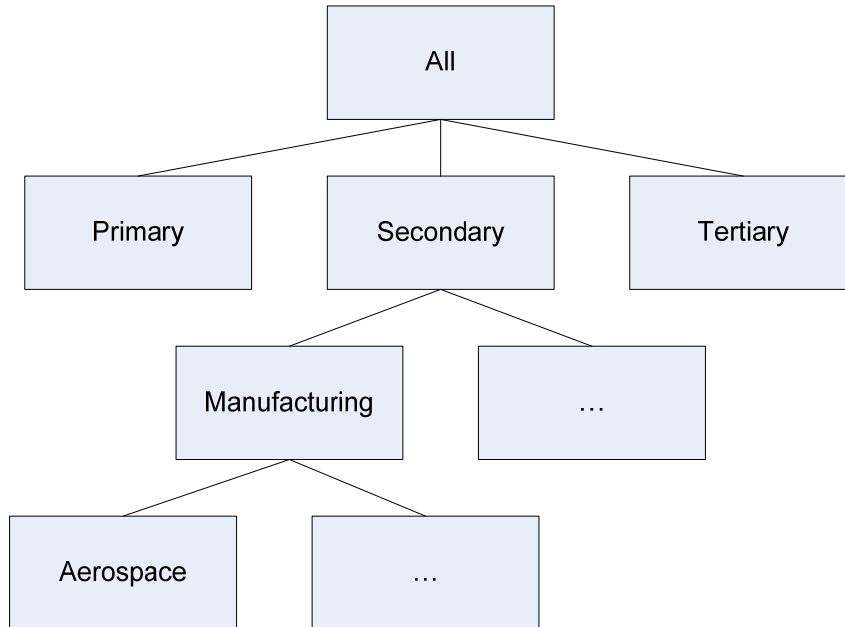
Figure 4.2: Sample hierarchy of industries

Figure 4.2 shows one way to divide the industries into a logical structure. The hierarchy is not fixed and if it seems that a different structure might be more appropriate, the hierarchy can be changed from the administrative interface.

Table "standards": This is a hierarchy of standards. The reason for it being a hierarchy is to allow for a more logical presentation of the standards when the user is asked to pick standards. Standards are also linked to:

- the geographical region in which they apply;

- the body to whom the standard belongs; and

- the industries to which they apply.

Figure 4.3 shows one way to divide standards.
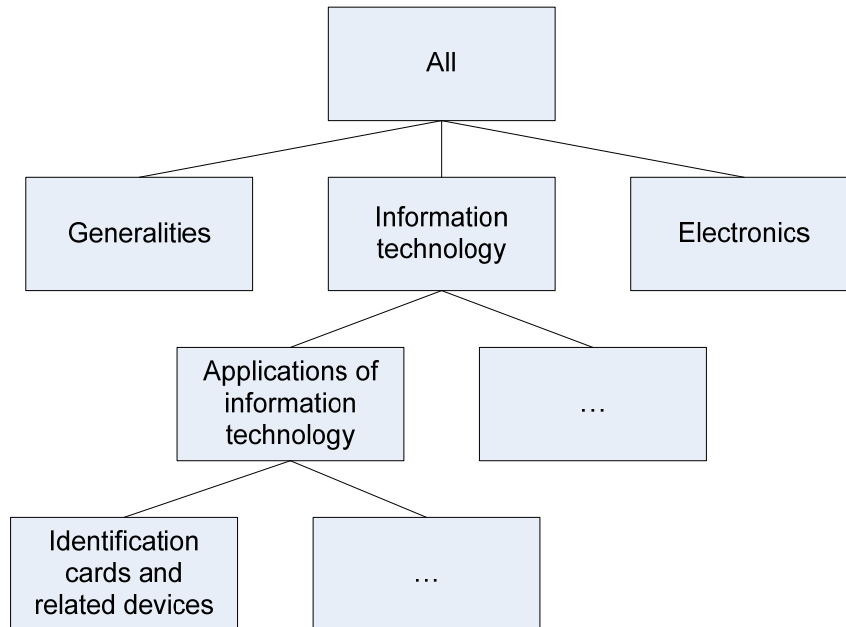
Figure 1.3: Sample hierarchy of standards

Table "interfaces": These are defined as the way in which a unit (a reader or writer) interfaces with the systems around it. An RS232 connection or Wiegand 24 connection is an example of this.

Table "factors": This is a hierarchy containing anything that might affect the default value for each of the 14 criteria. For example, a part operating at high altitude is likely to have a shorter operating life than the same part used only at ground level. Another example is that a metal part is likely to reduce the read and write range of a passive RFID tag. The factor is linked to:

- the criteria it affects;

- the technologies it affects; and

- the categories it affects.

In some cases the factors will be used to drive the reporting process of phase 3 (described in section 3.3.3.3). In other cases, the factors will modify the values of the criteria in one of the following ways:

- Reduce it to a fixed level

- Scale it by some factor

- Apply a user-defined function to the criteria

A more detailed description of how the factors will interact with the criteria values is given in section 4.2.2.
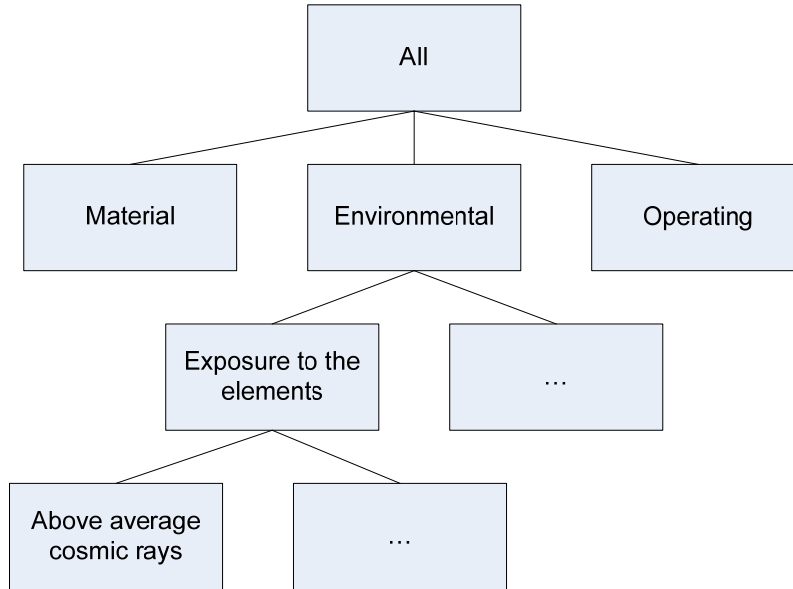
Figure 4.4: Sample hierarchy of factors

4.1.2. Processing information

Figure 4.5 shows a schematic view of the process for asking questions and the effect of their answers.
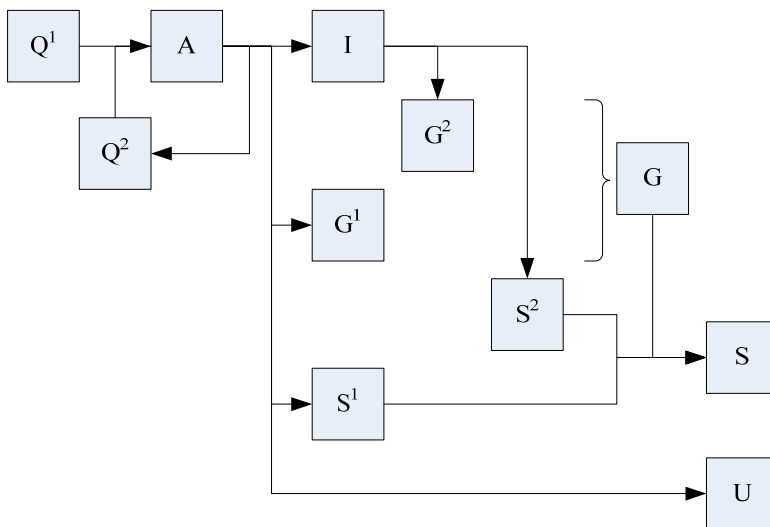
Figure 4.5: Phase 1 - information processing

Initially, the first set of questions Q^1 is retrieved.

Answers A will result in sets of:

- Industries I—in which the ID application will be used

- Geographies G^1—in which the ID application will be used

- Standards S1—with which the ID application must comply

- User criteria U—that the ID application should meet

- Further questions Q^2—that will be asked in a similar manner to Q^1

The loop Q2 > A will repeat until answer A yields no more Q^2.

At this point, I, G^1 and S^1 are further processed:

Industries I leads to:

- the geographies G^2 in which I operate; and

- the standards S^2 with which I must comply.

Geographies G^1 and G^2 are combined to form G, a list of all the geographical regions in which the ID application must work.

Standards S^1 and S^2 are added to the standards linked to geographical regions G to form S, the complete list of all the standards with which the ID application must comply.

4.1.3. Output of phase 1

The two outputs are:

- the list of standards S; and

- the set of filter values for the user criteria U.

4.2. Design of phase 2

4.2.1. Database information

As discussed earlier, for all the technologies in the database, the tool has functions to calculate the value of each criterion for a given technology. These functions draw on physical characteristics, such as length, width and height of the tag as well as non-physical characteristics, such as maximum number of bits stored by a barcode of a particular type.

4.2.1.1. Media

Media (stored in table "media") count as the physical objects that are in some way storing information; examples being a PVC credit card with a magnetic stripe, a particular RFID tag or a Contact Memory Button (CMB).

All media contain some physical descriptions of themselves as well as links to other information.

The internal physical descriptions (stored in table "media") are as follows:

- Nominal life: Expected life in months given "normal" operating conditions

- Surface impact: Damage, if any, done to the surface of the part being tagged, by the chosen tag, e.g. sticking a label onto a part has minimal surface impact

The linked physical descriptions are as follows:

- The object description (table "objectdesc") contains the weight of the media and a link to the second part, its dimensions.

- The physical dimensions (table "dimdesc") store the dimensions of the media. There are two options for approximating the dimensions of the media:

    1. as a rectangular base, storing the length and width of the rectangle and storing the height separately; or

    2. as a circular base, storing the diameter of the circle and storing the height separately.

- The characteristics of the media (table "chars") stores:

    1. Operating temperature range: The temperature range over which the device must be fully functional

    2. Extreme temperature range: The media will not be destroyed by these temperatures, but may not be fully functional

    3. Operating humidity range: The humidity range over which the device must be fully functional

    4. Extreme humidity range: The media will not be destroyed by these humidity conditions, but may not be fully functional

Note: The characteristics stored are for the material of the media only. When considering the actual operating temperature range, all the sub-components of the media must be considered. If the limiting factor is the RFID chip embedded in the media, then that will dictate the actual operating temperature range of the media.
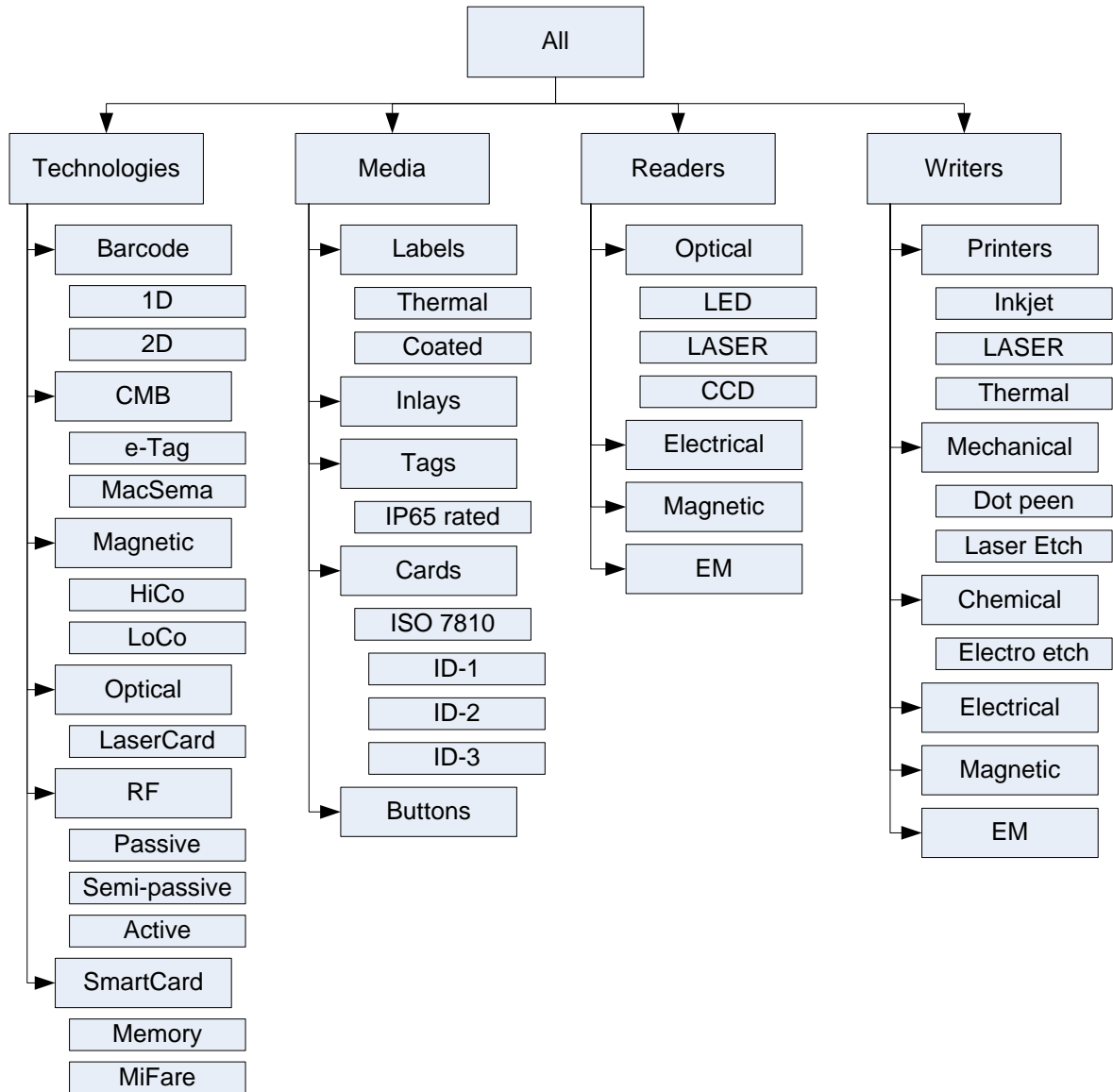
Figure 4.6: Sample of category hierarchy

The purpose of each level of the hierarchy, shown in Figure 4.6, is two-fold. Firstly, by assigning media, readers and writers to these categories, it will be easier to present them to the user in a more logical and structured format. Secondly, entries in table "factors" can be linked to any of the categories in the hierarchy, indicating that that factor operates on all the elements in that particular category. The process is detailed in the following section.

4.2.2. Processing information

To summarize, the information stored in the database is as follows:

- the physical properties and characteristics of the media and units (both readers and writers);

- the properties of the technologies (*media[2], *readers, *writers);

- the properties that relate media to readers (*encoding, *interface); and

- the factors that affect the different categories.

Each time the criteria filter interface is refreshed, the tool recalculates the value for each of the 14 criteria for each of the media configurations. Depending on the level of detail selected in the interface, the values for each media configuration can be aggregated:

- once, to form a range of values for that media as whole;

- twice, to form a range of values for the category to which that media belongs; and

- as many times as required, to simplify the number of categories being displayed.

Following is a pseudo-code explanation of the process.

---

[2] The * is used to represent any and all of the technology abbreviations (bc, cmb, mag, opt, rf and sc)

Line:

1   Retrieve the list of mediaconfigs

2   Retrieve the list of 14 criteria

3   For each mediaconfig:

4   Use links to retrieve the media

5   Use links to retrieve the physical dimensions and weight

6   Use links to retrieve operating and extreme characteristics

7   Use links to add information associated with each technology for that configuration

8   For each technology:

9   Use encoding and interface relationships to retrieve list of possible readers

10   Remove readers that do not meet with **standards S** from phase 1

11   For each criteria:

12   Retrieve function that calculates a value for the criterion based on that mediaconfig's values for that technology.

13   If necessary to calculate criterion value, retrieve encoding methods.

14   If necessary, remove encoding methods that do not meet **standards S**.

15   Apply retrieved function to calculate criterion value.

16   Retrieve factors that link the current criterion with the categories to which the media, technology, readers and writers used in the above function belong.

17   Apply each factor, in turn, to the calculated criterion value.

18   Store the resulting value.

19   Calculate the list of categories to be displayed, based on the user selected detail level.

20   For each category in the list:

21   Aggregate as necessary each stored value, for each technology, for each media configuration, for each media, for each sub-category beneath the current category.

22   Return the category list with aggregated minimum and maximum values for each criteria.

4.2.3. Output of phase 2

After the user has been through several cycles of refining the values for each of the criteria, we assume that the user will be left with a reasonably small number of media configurations remaining that still meet all the user's requirements. This list is passed on to phase 3.

4.3. Design of phase 3

4.3.1. Database information

Nearly all of the information for this phase is drawn from the tables introduced in the previous two phases. The two exceptions are detailed below:

- Table "companies" contains the company name, the contact information and a link to the company's website.

- Table "links" contains a list of links to information. Each link can be to a website or a local file. Most of the media, media configurations, readers, writers, units, companies, encodings, interfaces and standards will contain links that point to websites with more information about that element.

4.3.2. Processing information

The tool generates the report for the user by taking each of the remaining media configurations from phase 2 and summarizing the information as well as adding the related manufacturers' information and website links.

4.3.3. Output of phase 3

The output of phase 3 is a printable report.

4.4. Design of companies and users

4.4.1. Database information

As outlined in section 4.3.1, table "companies" stores information about companies. However it is not just manufacturers who would have entries in the table. The table will also contain:

- standards bodies;

- research institutions;

- members of the Aero-ID programme; and

- any company that requires access to the tool.

Table "users" will store:

- the user login name;

- the password;

- the level of the user (administrator, regular, etc);

- an email address;

- the date on which the user was created; and

- a link to the company to which the user belongs to.

Table "companyarea" a list of functions that a company might be involved with. Used primarily out of interest for the administrators, it includes areas such as:

- research

- manufacturing

- consultancy

- Auto-ID member

- chip design

- standards body

4.5. General administration

As explained in section 3.3.2.2, one of the roles of the administrator is to moderate the information entered. This is done by attaching some information to each new entry, recording which user made the entry and when it was made.

4.5.1. Database information

Table "entrymgmt". short for entry management, contains:

- a link to the user that made the entry;

- the date on which the entry was made;

- a link to the user that validated the entry; and

- the date on which the entry was validated.

4.5.2. Processing information

The process of moderating the entries involves searching for and displaying every entry that has not been validated; and then allowing the administrator to view the entry and validate it, if they feel it is suitable.

5. Implementation

This chapter explains each of the 14 criteria used by the tool and then goes on to describe the future developments for the tool. It also presents the feedback from the meeting with Cliff Fowkes (section 5.2).

5.1. Description of criteria

The following sections describe each of the criteria on which the user can filter technologies. The titles are mostly self-explanatory, but each criterion contains an explanation of how it is calculated for the different technologies. The filter threshold type is also noted for each criterion (see section 3.2.2 for further explanation).

5.1.1. Data storage (bits)

Filter threshold type: "more than"

The values displayed on the graphs will be the maximum values for that media configuration.

Barcode: It takes the best method of encoding (given standards constraints) and calculates the amount of data that could be written using the greatest number of dpi to print on the area of media available.

CMB, Magnetic stripes, Optical Stripes, RFID and SC: They take the sum of all the user-writeable areas of data to be the total data storage.

Direct Part Marking (DPM): It takes the best DPM writer and the best encoding and calculates how much data could be stored in the area that is available.

5.1.2. Read range (mm)

Filter threshold type: "between"

As a criterion, read range is one of the criteria most susceptible to the subtleties of the environment in which the ID solution is being used. The threshold is set to "between", allowing the user to specify both a minimum and a maximum read range. The maximum range is, perhaps, for privacy reasons.

Barcode: The outside limit is given by the focal range of the best reader (given standards constraints). However, depending on the resolution of the barcode, the reader may not be able to read that given barcode over the whole of its focal range. An improvement is suggested in section 5.3.8.

CMB, Magnetic stripes, Optical stripes and SC: They all have a read range of 0mm.

RFID: A relationship has not yet been identified. Some suggestions are made in section 5.3.8.

5.1.3. Simultaneous reads (number)

Filter threshold type: "more than"

This criterion describes the number of media that can be read simultaneously with one "read" of the reader. There are some unexpected results, such as a CMB reader that has two CMB interfaces and can read both at once.

Barcode, CMB, Magnetic stripes, Optical stripes, RFID and SC: are all defined by the greatest number of simultaneous reads for the best reader (given the standards constraints).

5.1.4. Read time (seconds)

Filter threshold type: "less than"

This is the time taken to perform a "read". Both simultaneous reads and read time must be considered together to calculate the overall number of media that can be read per second. We will illustrate how the same overall read rate could be achieved for very different situations and therefore why read rate was not chosen as the criterion.

On a fast moving conveyor belt each item might be passing through the reader's read area very rapidly. The reader only needs to be able to read one media at a time, but it has a very limited time in which to perform the "read".

At the other extreme, a warehouse rack might contain a thousand items. If they are not moving, then, within reason, the read time is not limited and so it would be acceptable to read one hundred media simultaneously, in a time one hundred times as long as the conveyor belt reader.

The overall read rate is the same, but the solution for one application will not work for the other application.

Barcode: The best reader (given standards constraints) will dictate the average read time.

Magnetic stripes: The best reader (given standards constraints) will dictate the fastest speed at which the media moves through the reader. By dividing the stripe length by this speed, the read time is calculated.

CMB, Optical stripes, RFID and SC: The best readers (given standards constraints) give a data read rate. The user criteria for the amount of data storage required can be divided by the read rate to give a read time.

5.1.5. Read area (mm^2)

Filter threshold type: "more than"

The read area is the area, in a plane perpendicular to the axis of the reader, in which an ID will be acquired from a media. Where the user sets the threshold, it will be dictated by the amount of control the user has over where the ID media is placed. If the media can be guaranteed to be in the same place on every product and the product can be guaranteed to be in the same orientation and position when it is read, then the required read area would be 0 mm^2. The greater the uncertainty or lack of maintaining order, the greater will be the required read area.

CMB, Magnetic stripes, Optical stripes and SC: All these must align exactly with their readers and therefore have 0 mm^2 read areas.

Barcode and RFID: The best reader (given standards constraints) will dictate the read area through its scanning angle or antenna. Further suggestions are made in section 5.3.9.

5.1.6. Life expectancy (months)

Filter threshold type: "more than"

The life expectancy figure should be the nominal life over which the media will perform whatever functions are expected of the media. Like read range, this criterion will be strongly influenced by the environment in which the media is used.

Barcode, CMB, Magnetic stripes, Optical stripes, RFIF and SC: For now, the figure is just a nominal life for the media. Eventually it should be the worst life for all the sub-components of the media e.g. the worst life of the RF chip, the antenna, the chip-antenna connection, the physical media etc.

5.1.7. Constant transverse speed (mm/s)

Filter threshold type: "between"

The constant transverse speed limit is defined as the fastest speed, perpendicular to the reader, at which the reader is still capable of reading the media.

Barcode, CMB, Magnetic stripes, Optical stripes, RFID and SC: For now, the tool makes a very rough approximation to the maximum constant transverse speed. By taking the width of the read area (section 5.1.5) and dividing it by the read time (section 5.1.4) an upper limit for the constant transverse speed is calculated. Future improvements for this calculation are suggested in section 5.3.11.

5.1.8. Character set (bits/character)

Filter threshold type: "more than"

For several technologies, the readers output bytes, in which case the character set is said to be "byte" or 8 bits/character. For readers that emulate other readers, for example, a barcode reader that connects to the computer and emulates being a keyboard, the range of characters they can produce is limited. There are specifications that define which characters can be stored. As a generalization, they can be broken down into numeric, single-case alphanumeric, multi-case alphanumeric, ASCII and full Unicode.

 Barcode and Magnetic stripes: They are defined by the encoding standard.

CMB, Optical stripes, RFID and SC: They can read byte streams or 8 bits/character.

5.1.9. Authentication to read data (cryptographic level)

Filter threshold type: "more than"

We have defined authentication to read data as the level of cryptographic protection required to access the user-stored data. This does not include how the user chooses to encrypt the information before they write it to the media. To illustrate this, consider a barcode — all the information is displayed and anyone with a barcode reader can access the information. The

authentication to read data is therefore "none", even though the information actually might be encrypted and require decrypting before it is of use.

Compare this with a MiFare RF media, whereby the reader must authenticate itself with the media by means of a cryptographic key *before* any of the information stored on the media can be read. In this case, the authentication to read data is "SHA1" or whatever the name of the cryptographic mechanism used to protect the information is. It is possible that once the information is retrieved from the media, it, too, like the barcode, may need further decrypting to be of use.

Barcode, CMB, Magnetic stripes, Optical stripes, RFID and SC: The scale has not yet been fully defined and improvements are suggested in section 5.3.12.

5.1.10. Humidity (RH) and temperature (^{0}C)

Filter threshold type: "between"

For all the four characteristics — Operational humidity, Operational temperature, Extreme humidity and Extreme temperature — the user can specify the range within which the media must survive. "Operational" characteristics refer to the range over which the media must be fully functional. "Extreme" characteristics refer to the range over which the media will not be destroyed, but might not be fully functional.

Barcode, CMB, Magnetic stripes, Optical stripes, RFID and SC: As with life expectancy (section 5.1.6), for now the tool uses the characteristics of the media to give the ranges, but in future, the ranges should be given by the overlapping region of the ranges of all the sub-components of the media.

5.1.11. Physical area required (mm^2)

Filter threshold type: "less than"

Barcode, CMB, Magnetic stripes, Optical stripes, RFID and SC: The physical area required for the media will be a characteristic of the media.

Direct Part Marking: The area will be a constraint of the part being marked and will be determined in phase 1, the question-answering phase.

5.2. Feedback

The tool was demonstrated to Cliff Fowkes, from MBDA, Stevenage. As a potential user of the tool, we felt that he would be able to give valuable feedback.

With respect to the operations of MBDA, Cliff Fowkes felt that the organization should try to understand an identification technology and then find applications for it, rather than trying to match the right technology to a specific application. He said that MBDA was likely to do more of the latter in the future and that a tool such as this would be helpful if it could meet the criteria discussed below.

For MBDA, the most important thing when dealing with specifications is to validate all the specifications, to ensure that they are operating within specification. The results of the experiments to confirm the specifications may either show the manufacturers to be under-generous, spot on or over-generous. In the latter case, MBDA would rather keep this information in-house so as not to risk libel. A method for achieving this validation is outlined in the following improvements section 5.3.1.

Cliff liked the visual process, comparing it favourably with other multi-variable optimization tools that he had used. He suggested making use of colour gradients to indicate a "comfort zone" for each criterion, within which the technology was sure to perform. This would blend out towards the edges of the criterion, indicating a pushing of the technology. This is further described in section 5.3.3.

5.3. Improvements

This project is a working demonstration that should provide the basis for the future of the application matching tool. The code has been written in a way that it should be possible to continue development on the existing code rather than starting from scratch. In the following sections, we present improvements that are not currently implemented.

5.3.1. Future design of phase 4

In order to allow for the overriding of the default values and relationships and entering experimental data as described in section 3.3.3.4, the following changes would be necessary.

Each many-to-many table in the database would need to be linked to a unique identifier. The default relationships would all have the same identifier. As companies want to override the values, a new identifier is created for that experiment for that company. Where the old

relationship is used, the tool would then use the new identifier or perhaps offer the user the choice of which identifier to use. For the one-to-many relationships, they would first have to be turned into a form similar to a many-to-many relationship, i.e. using a joining table to make the relationships. They could then be treated in the same way as many-to-many tables described above.

5.3.2. Storing sessions

A session management system could be implemented so that when users log on, they can either resume a previous session or start a new session. The session would store the answers for all questions and the values for the filters for each of the 14 criteria. This would allow the users to continue from where they left.

5.3.3. Comfort zones

As suggested by Cliff Fowkes, coloured gradients could be used to indicate where on a normal distribution type curve the following regions lie: "comfort zone", "stretching it" and "pushing it to the limit".

5.3.4. Criteria review

The 14 criteria identified so far do not make up an exhaustive list. Further input from companies that have faced the ID technology selection process would help add, modify or remove criteria from this list. An example of a criterion that might be needed in the future is the ability of the media to withstand processes, such as particular cleaning fluids or reflow soldering.

5.3.5. Factor review

As more experimentation is done into all these technologies, there will be new factors that are identified as affecting the performance of the criteria. These can be added into the system relatively easily.

5.3.6. Database driven factors

At the moment, the effect of the factors on the criteria values is only at a mock-up stage to demonstrate their effect. In order to properly implement them, a processing function will need to be written, that performs, as outlined in section 4.2.2, pseudo-code lines #16 and #17.

5.3.7. Separation of data storage area from ID area

As explained in the data storage criteria (section 5.1.1), the tool does not distinguish between the different types of user-writeable data storage areas. It would be desirable to extend the tool to be able to distinguish between types of data area, as well as the capability of storing on-chip functionality, such as on-board Health and Usage Monitoring Systems (HUMS).

5.3.8. Read range

For barcode readers, the reader will have an angular resolution. The angular resolution will dictate the maximum number of dpi (dots per inch) it can read at a given distance. By using the encoding method to calculate the number of dpi required to encode the desired amount of data in the desired space, the read range over which that dpi can be read can be calculated.

For RF readers, further investigation might yield a relationship between the media antenna, the reader antenna and some characteristics of the media and the reader, and the overall read range. This would provide a much more accurate read range, than the approximation used at the moment.

5.3.9. Read area

For barcode readers, the read area is usually described in terms of two angles on a vertical and horizontal axis over which the reader will read. Often the characteristics vary over this range; and it needs further investigation to see whether the variation could be incorporated into the tool. It might also be possible to store the maximum angle, normal to the reading plane, at which a barcode can still be read.

For RF readers, the antennae are quite often described by polar field-strength graphs. With further investigation, it might be possible to capture this information and make use of it when indicating how likely the reader is to achieve the stated read-range.

5.3.10. Life expectancy

With further experimentation on each media configuration, it may be possible to determine the life expectancy of each sub-component of a particular media configuration. In this way, the stated value for life expectancy would be the worst of all the sub-components. This would make the criterion more accurate.

5.3.11. Constant transverse speed

As with read range, further investigation is needed to determine a relationship between a given media configuration, a reader and the maximum constant transverse speed at which the media can move with the reader still making an error-free read.

5.3.12. Authentication to read data

Further investigation of the different technologies is needed to determine what levels should be put on the cryptographic scale.

5.3.13. Miscellaneous table implementation

Table "EDAC" is in the database schema as a table that would, in some way, store the Error Detection and Correction ability of encoding methods and interfaces. This might form another criterion, perhaps entitled, "criticality of accurate data reading".

Table "*writers", mentioned in section 4.2.1 are not yet used. We envisaged that at some stage in the future, the tool would have the ability to present a suggestion for a complete automated ID solution, down to which media, readers and writers would work together. The focus of the project so far has been on media and readers properties and characteristics.

Table "counterfeit measures" is in the database schema as a table that would, in some way, store the counterfeit measures associated with a particular media. Examples identified include micro-text and customized holograms that make a label or PVC card difficult to copy. The table is unused at the moment.

5.3.14. Distinction between open and proprietary standards

In our interview with MBDA, Cliff Fowkes suggested that his first choice for an ID technology would be an existing standard that was not proprietary, if it met all his other requirements. By adding a field into every table that represents something that might be proprietary intellectual property, the ability to filter out proprietary solutions could be added.

5.4. Technical details

5.4.1. Software tools

A functioning demonstration has been created that includes nearly all of the functions described in section 3. The following technologies have been used to build the web-based demonstration:

PHP[3] drives the dynamic web pages and performs all the queries on the database. Zend Development Environment has been used to code the PHP pages. The web pages can be found in the web-server root folder (/) and the administration (/admin/) folder.

MySQL[4] is the SQL database used. DBDesigner 4[5] by fabForce helped in the designing of the entity relationship (ER) model. The database is compatible with MySQL version 4 and upwards.

Xampp[6] by Apachefriends was used to provide a testing server environment on a Windows XP platform. The XamppLite version includes Apache web server, PHP 4, MySQL 5 and phpMyAdmin.

Macromedia Flash[7] facilitated in the programming of the main user interface of the tool. In order to pass information between the MySQL database and the Flash interface, Macromedia Remoting Connector and amfPHP was used (described below).

The Macromedia Remoting Connector[8] allows calls to functions in other programming languages to be made from within the native ActionScript.

amfPHP[9] allows the Macromedia Remoting Connector to specifically call PHP functions. Using this, most of the processing was performed on the database information in PHP and

---

[3] http://www.php.net/
[4] http://www.mysql.com/
[5] http://fabforce.net/dbdesigner4/
[6] http://www.apachefriends.org/en/xampp.html
[7] http://www.adobe.com/products/flash/flashpro/
[8] http://www.adobe.com/support/documentation/en/flash_remoting/fl8/releasenotes.html
[9] http://www.amfphp.org/

then the results were returned directly to the ActionScript. The PHP code for these functions is in the amfPHP services folder (/flashservices/services/autoID.php).

Macromedia Dreamweaver[10] helped to make the web pages and manage the synchronization of the pages between the development machine and the demonstration server.

The ImPAKT[11] tools by InterAKT helped generate some of the PHP, particularly on the administration pages.

The QuB3 tool included in ImPAKT by InterAKT helped to visually generate the SQL queries, particularly for the queries that drive the Flash interface.

phpMyAdmin[12] was used in conjunction with DBDesigner 4 to enter test data into the database.

5.4.2. Hierarchical tree structures in the database

In order to understand how to work with the hierarchies in the database, such as table "categories", the following section outlines how the 'tl' and 'tr' fields are used to create a tree.
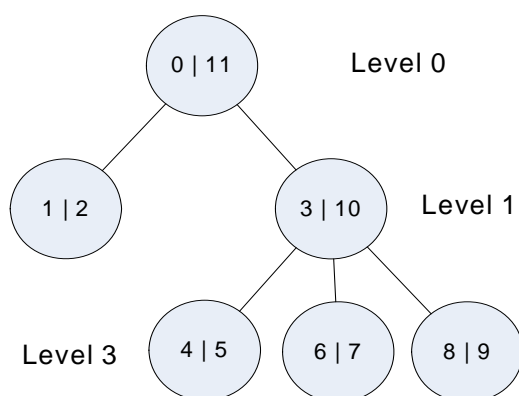
Figure 5.1: Sample tree

The sample tree (Figure 5.1) shows a tree with three levels. Within each circle, 'tl' is the left figure and 'tr' is the right figure.

As you can see, 'tl' for the first child of a parent, is always the parent's 'tl' + 1. For example, 4|5 is the first child of 3|10, and 4 is 3 + 1.

'Tr' for the parent is always 'tr' of the last child + 1. For example, 3|10 is the last child of 0|11 and 11 is 10 + 1.

This leads to some interesting and useful properties of the tree.

- The leaves of the tree can be identified by the entries where 'tr' = 'tl' + 1.

- The path to a given entry can be found by searching for 'tl' < 'tl$_{entry}$' AND 'tr' > 'tr$_{entry}$'.

Deleting and inserting entries does not require recursive updating of 'tl' and 'tr' values. Each entry has two, unique numbers—one for 'tl' and one for 'tr'. Deleting a leaf involves subtracting 2 from affected entries, whilst inserting a leaf involves adding 2 to affected entries.

---

[10] http://www.adobe.com/products/dreamweaver/
[11] http://www.interaktonline.com/Products/Bundles/ImpAKT/Overview/
[12] http://www.phpmyadmin.net/

The effect of deleting and inserting leaves on 'tl' and 'tr' values is shown in Figure 5.2 (a) and (b).
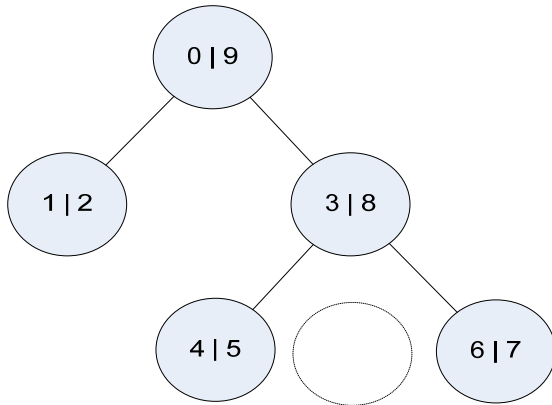
Figure 5.2 (a): Example of deletion
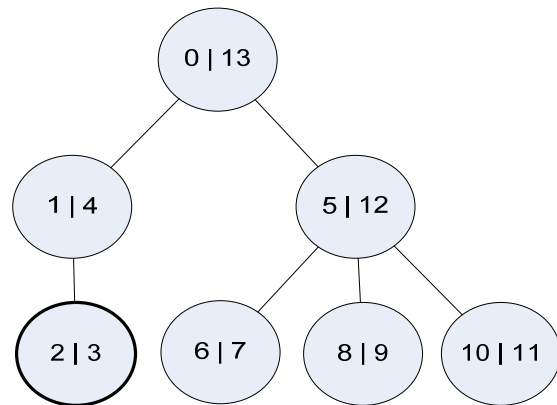
Figure 5.2 (b): Example of insertion

The process of moving entries from one parent to another is more efficient than simply deleting all the entries from the one parent and inserting them under the new parent. Since an operation can be applied to a whole block to entries between a range of 'tl' and 'tr' values, it is possible to:

- "Move" the entries out of the tree

- Close up the gap

- Create a new gap

- "Move" the entries back in

The process is outlined in Figure 5.3.

I have written the SQL queries to perform these operations and they can be found in the PHP files in the /admin/ folder of the website. Using these routines, the following hierarchies are maintained:

- Factors

- Categories

    1. Technologies

    2. Media

    3. Readers

    4. Writers

- Industries

- Geographies

- Standards
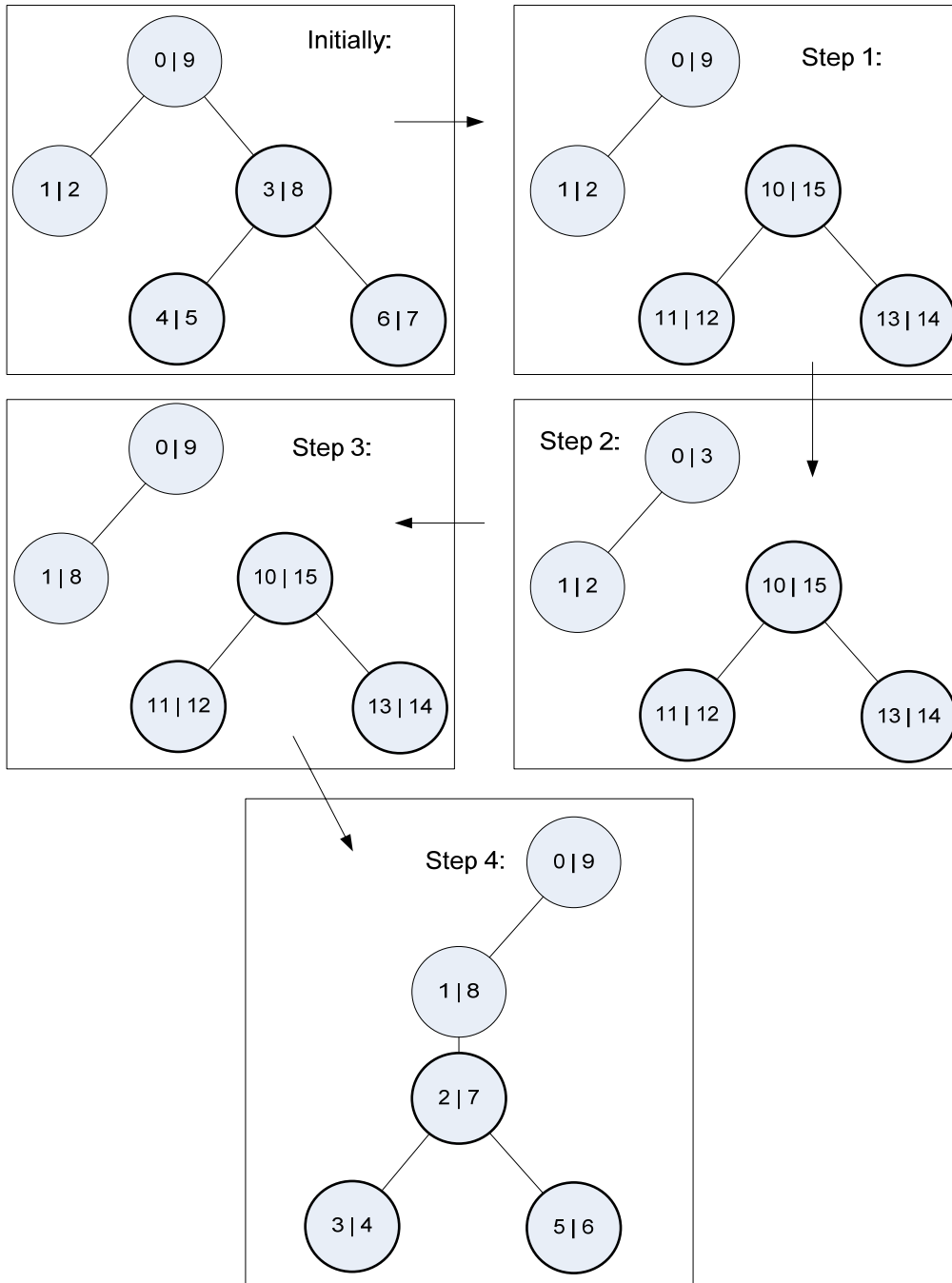
Figure 5.3: Four-stage process to move an entity

6. Conclusions

The main conclusion is that it is possible to characterize the majority of the criteria on which technology selections are made. Many of the characteristics of an overall ID solution will depend upon the exact environment in which they will be used. The tool does not model the environment and therefore the functionality of phase 4, putting investigation results back into the tool, is essential in making the tool relevant to the user's environment.

The design philosophy used in the creation of this demonstration of an ID technology selection application is such that if the Aero-ID group is interested in pursuing this tool, they will be able to extend the demonstration to add more functionality, rather than having to start programming from scratch.

For the tool to be of use, it is clear that it must be kept up-to-date. This means that it must contain the latest:

- Media

- Encoding methods

- Interface protocols

- Standards

- Readers

- Writers

- Relationships between environmental factors and the 14 criteria

This would require a continual effort from manufacturers inputting their latest products. There is an incentive for the manufacturers to enter their products into the system. However there is not the same motivation for standards bodies to input their new standards into the system. This means that the tool administrators will probably have to make sure that as new standards are created, they are entered into the database.

The web-based tool would be an acceptable method of delivery. However companies would want to be reassured that the experimental data they entered into the system would not be shared without their consent.

It is very difficult to precisely capture an environment and all the factors that affect ID technologies. This tool will be limited in absolute accuracy unless it includes 3D models of the part and the environment and so on. The approximations it can offer will still save a significant amount of time for someone tasked with selecting the correct ID technology for their application.

Appendix A. Attached Files

After extracting the zip file containing all the files, there will be the following files:

- appmatching.fla

- criteriaDescriptions.fla

- sortlisttest.fla

- Draft Vx.xx.xml

- Draft Vx.xx.sql

- autoID vx.xx.zip

The following appendices describe what these files contain.

A.1. PHP Code

The PHP files are within the zip file autoID vx.xx.zip. Upon extracting this file, there will be two folders:

- autoID; and

- flashservices

autoID contains the PHP for the web pages as well as the files used by InterAKT. The most important file is:

- /autoID/Connections/autoID.php

This stores the details of the sql server to which all the other pages will connect. The important lines of code are:

```
$MM_autoID_HOSTNAME = 'localhost';

$MM_autoID_DATABASE = 'mysql:test';

$MM_autoID_USERNAME = 'autoid';

$MM_autoID_PASSWORD = 'diotua';
```

flashservices contains the PHP for the Flash remoting connector. The most important files are:

- /flashservices/gateway.php—storing the configuration options for the remoting

- /flashservices/services/autoID.php—storing the code for the remoting interface

In autoID.php there are the following lines of code:

```
var $dbhost = "localhost";

var $dbname = "test";

var $dbuser = "root";

var $dbpass = "";
```

These should be set to the same values as the other autoID.php connection.

If you have not used amfPHP before, you should note that after changing the functions in autoID.php you should browse to:

 http://yourwebserver/flashservices/browser/methodTable.php?class=autoID

This will update the method table for the autoID.php connector. If you do not do this, the remoting will probably not work.

A.2. Database schema

The schema is contained in the file Draft Vx.xx.xml. It can only be opened in fabForce DBDesigner 4.

A.3. SQL dump

A dump of the SQL database, suitable for recreating the database using a tool such as phpMyAdmin is available in the file Draft vx.xx.sql.

Running this query will create the structure, populate the values and add the constraint relationships. It is compatible with MySQL 4 and above.

A.4. Flash ActionScript

The Flash ActionScript is contained within the following files

- appmatching.fla
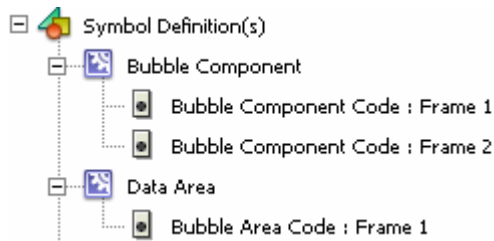
All the functional action script is contained within the three frames above. The application control code is in the Bubble Component, whilst the graphical display code is in the Data Area.

- criteriaDescriptions.fla
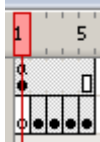
There is some functional actionscript in the top layer, however the function of this file is to provide the animations for the 14 criteria. These animations are located in frames 2–15.

Appendix B. Database Schema

The database schema is too large to fit on one page. The thumbnail below describes how the following pages should be arranged to see the whole schema.

Page 1

Page 2

Page 4

Page 3

rfwriter_has_unitconfigs
- rfmediawriter_idrfwriter: INTEGER (FK)
- unitconfigs_idunitconfigs: INTEGER (FK)

unitconfigs_has_rfreader
- unitconfigs_idunitconfigs: INTEGER (FK)
- rfmediareader_idrfreader: INTEGER (FK)

unitconfigs_rfreaders   unitconfigs_rfwriters

mediaconfigs_rfmedia

mediaconfigs_has_rfmedia
- mediaconfigs_idmediaconfigs: INTEGER (FK)
- rfmediamedia_idrfmedia: INTEGER (FK)

rfreaders_categories
categorys_rfreaders

categories_has_rfreader
- categories_idcategories: INTEGER (FK)
- rfmediareader_idrfreader: INTEGER (FK)

rfwriters_categories
categorys_rfwriters

categories_has_rfwriter
- categories_idcategories: INTEGER (FK)
- rfmediawriter_idrfwriter: INTEGER (FK)

unitconfigs_has_cmbwriter
- unitconfigs_idunitconfigs: INTEGER (FK)
- cmbwriter_idcmbwriter: INTEGER (FK)

unitconfigs_has_cmbreader
- unitconfigs_idunitconfigs: INTEGER (FK)
- cmbreader_idcmbreader: INTEGER (FK)

unitconfigs_cmbwriters

cmbmedia_categories
categorys_cmbmedia

mediaconfigs_cmbmedia

categories_has_cmbmedia
- categories_idcategories: INTEGER (FK)
- cmbmedia_idcmbmedia: INTEGER (FK)

mediaconfigs_has_cmbmedia
- mediaconfigs_idmediaconfigs: INTEGER (FK)
- cmbmedia_idcmbmedia: INTEGER (FK)

cmbreaders_categories
categorys_cmbreaders

cmbwriters_categories
categorys_cmbwriters

categories_has_cmbreader
- categories_idcategories: INTEGER (FK)
- cmbreader_idcmbreader: INTEGER (FK)

categories_has_cmbwriter
- categories_idcategories: INTEGER (FK)
- cmbwriter_idcmbwriter: INTEGER (FK)

unitconfigs_screaders

unitconfigs_has_screader
- unitconfigs_idunitconfigs: INTEGER (FK)
- screader_idscreader: INTEGER (FK)

unitconfigs_scwriters

unitconfigs_has_scwriter
- unitconfigs_idunitconfigs: INTEGER (FK)
- scwriter_idscwriter: INTEGER (FK)

scmedia_categories
categorys_scmedia

mediaconfigs_scmedia

categories_has_scmedia
- categories_idcategories: INTEGER (FK)
- scmedia_idscmedia: INTEGER (FK)

mediaconfigs_has_scmedia
- mediaconfigs_idmediaconfigs: INTEGER (FK)
- scmedia_idscmedia: INTEGER (FK)

screaders_categories
categorys_screaders

scwriters_categories
categorys_scwriters

categories_has_screader
- categories_idcategories: INTEGER (FK)
- screader_idscreader: INTEGER (FK)

categories_has_scwriter
- categories_idcategories: INTEGER (FK)
- scwriter_idscwriter: INTEGER (FK)

unitconfigs_bcwriters

unitconfigs_has_bcwriter
- unitconfigs_idunitconfigs: INTEGER (FK)
- bcwriter_idbcwriter: INTEGER (FK)

unitconfigs_bcreaders

unitconfigs_has_bcreader
- unitconfigs_idunitconfigs: INTEGER (FK)
- bcreader_idbcreader: INTEGER (FK)

bcmedias_catergories
categorys_bcmedias

mediaconfigs_bcmedias

categories_has_bcmedia
- categories_idcategories: INTEGER (FK)
- bcmedia_idbcmedia: INTEGER (FK)

mediaconfigs_has_bcmedia
- mediaconfigs_idmediaconfigs: INTEGER (FK)
- bcmedia_idbcmedia: INTEGER (FK)

bcreaders_categories
categorys_bcreaders

bcwriters_categories
categorys_bcwriters

categories_has_bcreader
- categories_idcategories: INTEGER (FK)
- bcreader_idbcreader: INTEGER (FK)

categories_has_bcwriter
- categories_idcategories: INTEGER (FK)
- bcwriter_idbcwriter: INTEGER (FK)

unitconfigs_magwriters

unitconfigs_has_magwriter
- unitconfigs_idunitconfigs: INTEGER (FK)
- magwriter_idmagwriter: INTEGER (FK)

unitconfigs_magreaders

unitconfigs_has_magreader
- unitconfigs_idunitconfigs: INTEGER (FK)
- magreader_idmagreader: INTEGER (FK)

magmedias_categories
categorys_magmedias

mediaconfigs_magmedias

categories_has_magmedia
- categories_idcategories: INTEGER (FK)
- magmedia_idmagmedia: INTEGER (FK)

mediaconfigs_has_magmedia
- mediaconfigs_idmediaconfigs: INTEGER (FK)
- magmedia_idmagmedia: INTEGER (FK)

magreaders_categories
categorys_magreaders

magwriters_categories
categorys_magwriters

categories_has_magreader
- categories_idcategories: INTEGER (FK)
- magreader_idmagreader: INTEGER (FK)

categories_has_magwriter
- categories_idcategories: INTEGER (FK)
- magwriter_idmagwriter: INTEGER (FK)

unitconfigs_optwriters

unitconfigs_has_optwriter
- unitconfigs_idunitconfigs: INTEGER (FK)
- optwriter_idoptwriter: INTEGER (FK)

unitconfigs_optreaders

unitconfigs_has_optreader
- unitconfigs_idunitconfigs: INTEGER (FK)
- optreader_idoptreader: INTEGER (FK)

optmedias_categories
categorys_optmedias

mediaconfigs_optmedias

categories_has_optmedia
- categories_idcategories: INTEGER (FK)
- optmedia_idoptmedia: INTEGER (FK)

mediaconfigs_has_optmedia
- mediaconfigs_idmediaconfigs: INTEGER (FK)
- optmedia_idoptmedia: INTEGER (FK)

optreaders_categories
categorys_optreaders

optwriters_categories
categorys_optwriters

categories_has_optreader
- categories_idcategories: INTEGER (FK)
- optreader_idoptreader: INTEGER (FK)

categories_has_optwriter
- categories_idcategories: INTEGER (FK)
- optwriter_idoptwriter: INTEGER (FK)

media_categories

categorys_media

categories_has_media
- categories_idcategories: INTEGER (FK)
- media_idmedia: INTEGER (FK)

units_categories
categorys_units

categories_has_unit
- categories_idcategories: INTEGER (FK)
- unit_idunit: INTEGER (FK)