THE GEORGE WASHINGTON UNIVERSITY

DECISION ANALYSIS: OPTIMIZING CHOICES OF DATA MAPPING SOFTWARE

SUBMITTED TO THE FACULTY OF
THE SCHOOL OF BUSINESS AND PUBLIC MANAGEMENT
EXECUTIVE MBA PROGRAM

BY

DEAN E. SONDEREGGER

ASHBURN, VIRGINIA

CLASS OF 2002

Table of Contents

1. Background 1

2. Problem to be Analyzed 2

3. Approach 2

4. Analysis 6

4.1. Goal 6

4.2. Objectives 6

Usability 6

Performance 7

Maintenance 7

Feature Set 8

Operational Support 10

Cost 11

4.3. Priorities 11

4.4. Evaluation 12

4.5. Sensitivity 13

List of Figures

Figure 1. Sample Hierarchy 3

Figure 2. Pairwise Comparison Matrix 4

Figure 3. Decision Hierarchy 12

Figure 4. Synthesis with respect to Goal 13

Figure 5. Sensitivity Analysis 14

Figure 6. Gradient Analysis for Cost 14

List of Tables

Table 1. Alternatives for Decision Analysis 2

1. Background

Over 70% of newly originated residential mortgages are sold after the loan has been funded. The marketplace for sale of mortgages between financial institutions is known as the secondary mortgage market. Ultraprise Corporation provides an Internet based exchange, ULTRAPRISE.COM, for the purchase and sale of residential mortgages on the secondary mortgage market. The purchase process involves several steps, including product discovery, price discovery, due diligence, and transfer of data and documents.

As part of the due diligence and transfer processes, all documentation for the loan is physically shipped to the purchaser's facility. Documents are reviewed for accuracy; data is extracted from the documents and manually keyed into the buyer's processing systems. Due to the high rate of error associated with manual data entry, a quality control team typically reviews the data after it has been keyed. This process of keying and reviewing data is expensive in terms of time and staff.

It has long been recognized in the mortgage industry that this particular point of pain is unnecessary – the seller has already entered all of this data into systems in order to create the documentation. Unfortunately, there exists no standard for capture and transmission of data for the secondary mortgage market. This makes the electronic flow of data very costly.

2. Problem to be Analyzed

As an online marketplace, ULTRAPRISE.COM has the potential to capture data directly from seller systems and to transfer that data through to buyer systems. The proposed mechanism by which this data capture occurs is through the use of extract files. Most seller systems have the ability to extract loan data into a file format of some type. The key to successful electronic transfer is the ability to map and translate the contents of the seller extract file into a common (Ultraprise) file format.

In order to add this capability to ULTRAPRISE.COM, the Ultraprise development staff can develop the capability internally, or can purchase a tool from a vendor that provides this capability. Four vendors were identified that could provide a product that meets Ultraprise's data mapping needs. These vendors combined with the possibility of an in-house development effort are shown in Table 1 below and represent five alternatives for this decision analysis.

Table 1. Alternatives for Decision Analysis

Vendor	Product
Decade Systems	TEDI
Microsoft	SQL Server Data Transformation Services (DTS)
Data Junction	Data Junction
Mercator	Mercator
Ultraprise	Home grown development

The choice of an alternative has a serious implication for Ultraprise. The insertion of a new technology into ULTRAPRISE.COM (or the development of that technology from scratch) will require key development resources and will delay other enhancements to the ULTRAPRISE.COM service offering. As a venture financed startup, Ultraprise must conserve its resources wisely and cannot afford to revisit this decision in six months.

3. Approach

This analysis utilizes the Analytical Hierarchy Process (AHP) in the evaluation of each alternative. AHP provides a decision methodology process that focuses on the achievement of objectives. The approach can be broken down into three basic steps: problem description, evaluation, and sensitivity analysis.

The description or decomposition of the problem involves setting up the model framework. The model requires a single formulated goal, objectives supporting that goal, and potentially sub-objectives supporting each objective. Figure 1 below shows an example of such a hierarchy. All objectives and sub-objectives ultimately contribute to the goal. An exhaustive list of alternatives provides the decision points that are evaluated against this hierarchy. Upon completion of the evaluation, priorities will be derived for each alternative reflecting the degree to which the alternative satisfies the goal.

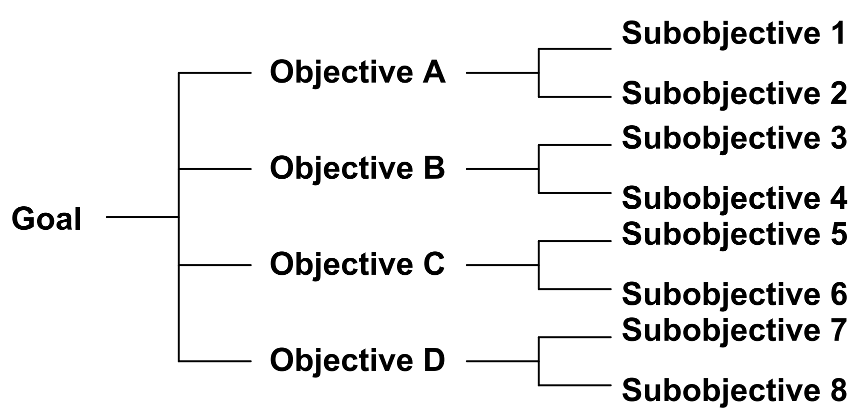

Figure 1. Sample Hierarchy

Once the framework is in place, the development of priorities takes place. Weights are assigned to each objective and sub-objective. These weights are assigned through a process known as pairwise comparison. In pairwise comparison, each objective is compared at a peer level in terms of relative importance. These comparisons can be done graphically, through verbal measures, or numerically. The tabulation of these comparisons creates a matrix as shown in Figure 2 below.

	A	B	C	D
Objective A		w_1	w_2	w_3
Objective B			w_4	w_5
Objective C				w_6
Objective D				

Figure 2. Pairwise Comparison Matrix

The weights associated with each comparison are calculated and after some linear algebra, a set of priorities for each objective is applied to the decision hierarchy. The careful observer will note that there is redundant information associated with the pairwise comparisons. When using AHP, the associative property (e.g. A=B, B=C, e.g. A=C) is not assumed to be true in all cases. This provides more accurate results, as the associative property is not true in all cases in the real world. This does however, lead to a level of mathematical inconsistency in the model, which is calculated separately as an output and must be addressed as part of any decision.

Once the priorities of each objective and sub-objective are established in the hierarchy, the actual evaluation of alternatives takes place. This involves yet another set of pairwise comparisons, this time between each alternative, evaluated against each objective and sub-objective. Once these comparisons are complete, each alternative will have a derived priority, representing how well that alternative satisfies the formulated goal.

Once the evaluation is complete, the results must be analyzed for sensitivity to different factors. The model lends itself to analysis of the sensitivity of the results to changes in priorities. This analysis is critical as it shows how changes in environment, priorities and other external factors will influence the decision process.

4. Analysis

4.1. Goal

The goal established for this analysis is to select a mapping software package (or to choose to build one in house).

4.2. Objectives

Through conversations with development staff, industry analysts, and executives at Ultraprise, six objectives have been established. The following sections explore each objective in detail.

Usability

Usability refers to how easy it is to use the software. The objective is for the software to be intuitive, easily learned and operated. Several sub-objectives were used to measure ease of use:

Complexity - Refers to the level of complexity of the user interface / command structure. The objective is to have a low degree of complexity while still maintaining a rich feature set.

Time requirements – Refers to the amount of time required to complete a data mapping exercise. The objective is to be able to complete data mappings in as little time as possible.

Stability – Refers to the propensity of the software to crash or experience spurious behavior. The objective is for high stability (and hence little/no crashing).

Performance

Performance refers to the operational performance of the mapping software. The desire is to have a high level of performance such that processing is rapid, has a minimal impact on the machine, and allows many simultaneous data translation activities. Several sub-objectives were used to performance:

Speed of Processing – Refers to the speed with which the mapping software performs a data mapping operation. The desire is for rapid processing.

Memory Usage – Refers to the amount of memory used by the software when loaded and processing a data map. The desire is to use as little memory as possible.

Multi-Threading Support – Multithreading refers to the ability for the software to handle multiple tasks at the same time[1]. The desire is to have robust multi-threading support.

Scalability – Scalability refers to the ability for the software to support many simultaneous mappings. In this context, scalability is measured as the ability to scale both within a single machine and across multiple machines. The desire is for a highly scalable application.

Maintenance

Maintenance refers to the set of activities required to keep the software running. The objective is to have software that requires little / no maintenance. Several sub-objectives were used to measure maintenance:

[1] From this description, this sub-objective could be referred to as multi-tasking, which is not the same as multi-threading. The objective requires multi-threading but the rationale for this need is beyond the scope of this paper. For the purposes of this paper, the terms will be considered to mean the same thing.

Reusability – Refers to the ability of mapping specialists to reuse data maps. Some software may allow for the creation of relationships between maps (parent/child behavior), the ability to create mapping libraries, or simply the ability cut and paste mapping information between maps. The desire is to have a robust reuse capability.

Ease of Administration – Refers to the difficulty of configuring and monitoring the software. The desire to have an intuitive, easily administered application that provides for automation of simple tasks.

Vendor Responsiveness – Refers to the speed and quality of vendor responsiveness to development and configuration issues. The desire is to have a service agreement that provides for robust support anytime, anywhere.

Portability – Refers to the ability to "port" the software (and associated maps, configurations, etc.) to a different operating system / environment. The desire is to have a portable application so that the company could shift to a Unix based system if desired to enhance scalability.

Ease of Upgrades – Refers to the impact that software upgrades have on the system. The ideal scenario would be one wherein the system doesn't need to shut down at all for minor upgrades.

Feature Set

Feature set refers to the functionality that exists in the software. The objective is for the software to be rich in features, providing many capabilities that would otherwise have to be developed from scratch. Several sub-objectives were used to measure the extent of the software feature set:

Interactivity – Interactivity refers to support for different means of interacting with the software. Types of interfaces that could be supported include batch, COM, and Corba. Additional considerations include the support for transactions as part of the interface. The desire is to have robust interface compatibility with different types of interfaces and with robust transactional support.

Error Handling – A key aspect of any enterprise software is the manner in which that software reacts to or handles errors. The desire is for robust error handling which means no errors will crash the system, errors are caught and reported on, documentation exists on common causes of errors, and that there exits a notification mechanism for critical errors to ensure adequate operational support.

Supported File Formats – The premise of the system is to map extracted data files from external systems. As such, the desire is for the software to support a broad variety of data formats to include various ASCII formats, Microsoft Excel files, XML file formats, direct connections to databases, and direct memory access. Additionally the desire is to support a number of vendor systems natively through default templates.

Manipulation Capabilities – The mapping software should be able to do much more than simply read and write data from a variety of formats. Critical features include the ability to manipulate the data through simple scaling operations, use of regular expressions (simple mathematics, string operations, etc), use of custom (user defined) expressions, and through the mapping to or from enumerated values. The desire is for a robust, extensible set of expressions and manipulation capabilities.

Supported Interfaces – Where possible, the software should be able to provide value beyond simply mapping and translating the data. As this data comes from and is delivered to external systems, the desire is for the software to support transport mechanisms and queuing interfaces such as HTTPS, MSMQ, MQSeries, and EDI interfaces.

Operational Support

Operational support refers to the support provided by the vendor for issues that may arise in a production environment. Several sub-objectives were used to measure the extent of operational support provided by the vendor:

Error Handling – A key aspect of any enterprise software is the manner in which that software reacts to or handles errors. The desire is for robust error handling which mean no errors will crash the system, errors are caught and reported on, documentation exists on common causes of errors, and that there's a notification mechanism for critical errors to ensure adequate operational support.

Diagnostics – When operational errors occur, the resolution of problems is a highly time sensitive matter. The desire is for extensive diagnostic capabilities to track down and help resolve errors before and after they occur.

Vendor Responsiveness – Refers to the speed and quality of vendor responsiveness to operational issues. The desire is to have a service agreement that provides for robust support anytime, anywhere.

Ease of Administration – Refers to the difficulty of configuring and monitoring the software. The desire to have an intuitive, easily administered application that provides for automation of simple tasks.

Cost

The expenditure of financial and human resources associated with the acquisition and maintenance of a software package impacts the desirability of the system. The desire is for the system to be inexpensive. Cost is measured in several manners that correspond to the following sub-objectives:

Fixed – Refers to the upfront investment required by Ultraprise. The desire is to have low upfront costs.

Maintenance – Software typically involves an annual license or support contract. The desire is to have low maintenance costs.

Transaction – Some software packages provide for transactional pricing. This would involve a fee for each use of the software (in this case each time a file was translated). The desire is to have no transactional component to the pricing.

4.3. Priorities

Each objective and sub-objective was evaluated through graphical pairwise comparison using the Expert Choice decision modeling software. The primary participant in the pairwise comparison was Dean Sonderegger, Vice President of Professional Services. Figure 3 shows the decision hierarchy and derived priorities of each objective / sub-objective.

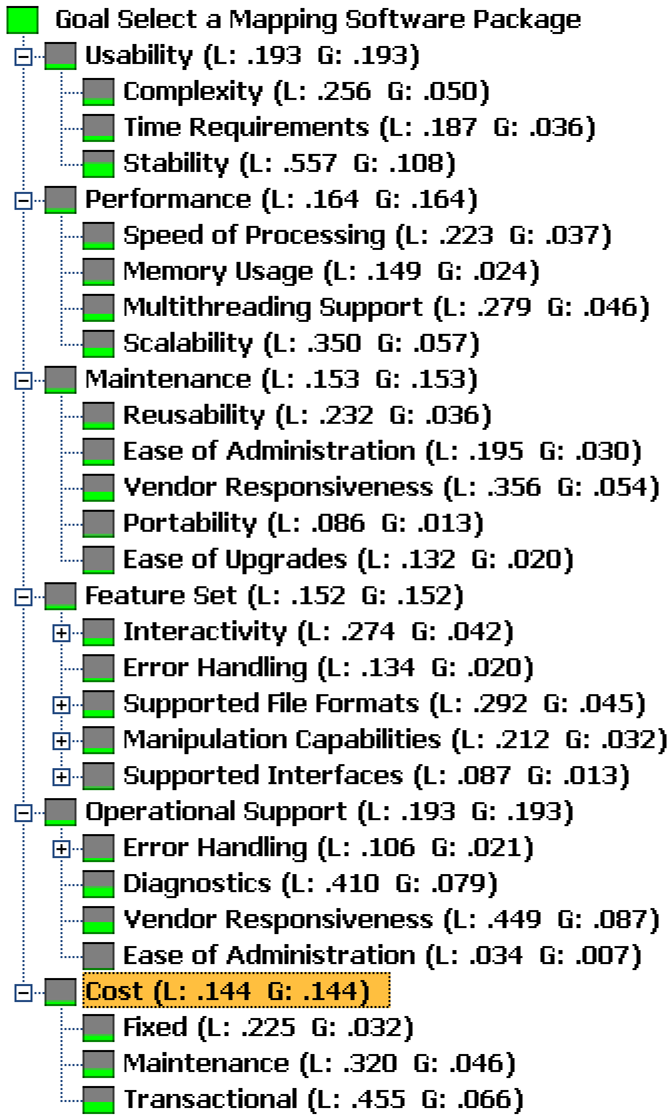

Figure 3. Decision Hierarchy

4.4. Evaluation

The five alternatives were evaluated against the decision hierarchy using graphical pairwise comparisons. Dean Sonderegger, Vice President of Professional Services, performed the comparisons with input from the development staff. The results of this evaluation are shown below in Figure 4 below. The preferred solution from the evaluation of the six objectives is to select Mercator's data mapping software. This is consistent with general impressions gathered through working with the various software packages.

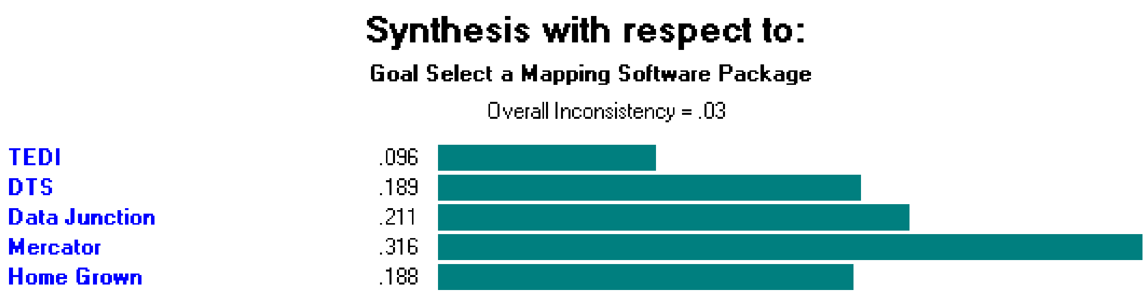

Figure 4. Synthesis with respect to Goal

4.5. Sensitivity

Figure 5 shows a sensitivity analysis of the three alternatives with respect to the six main objectives. From the pairwise comparisons performed as part of this analysis, it's apparent that the Mercator software is preferred for all objectives with the exception of cost. This is not surprising as the Mercator software package costs over $150,000 in upfront fees with a residual transaction fee of at least $1 per loan that passes through Ultraprise.

Figure 6 shows a gradient analysis of this scenario. When the priority of the cost objective is adjusted upward (just past .50), the optimal choice changes from Mercator to Data Junction. This reflects the substantial price difference between Data Junction and Mercator. Data Junction costs roughly $20,000, has an annual maintenance fee of

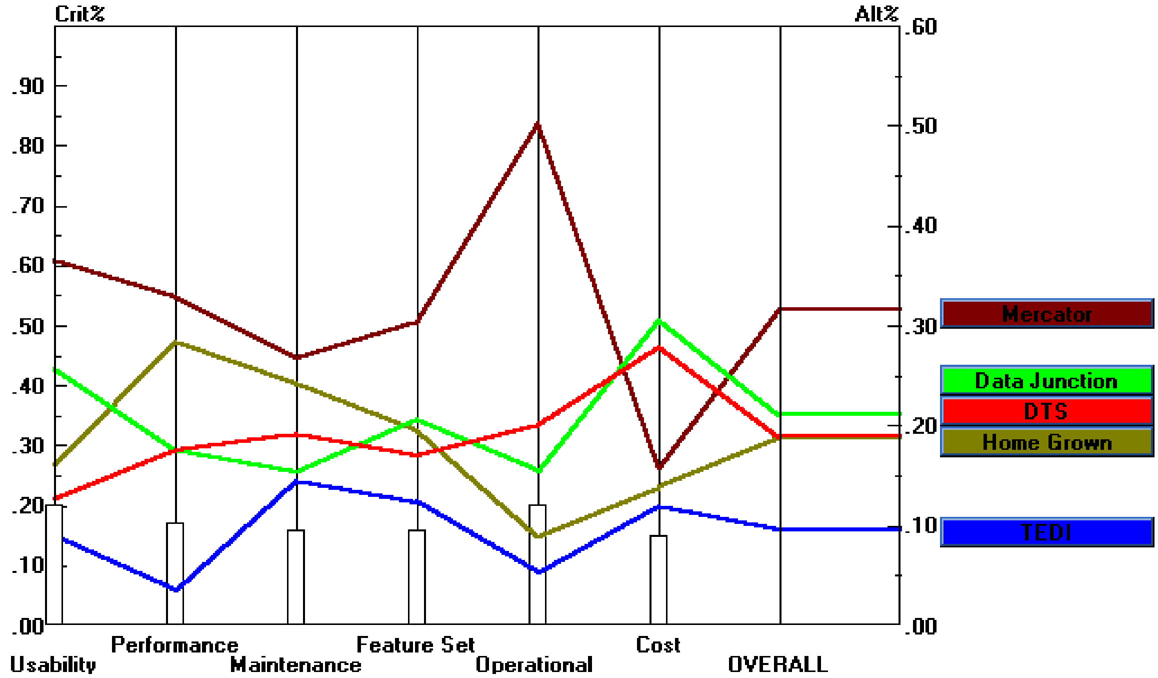

Figure 5. Sensitivity Analysis

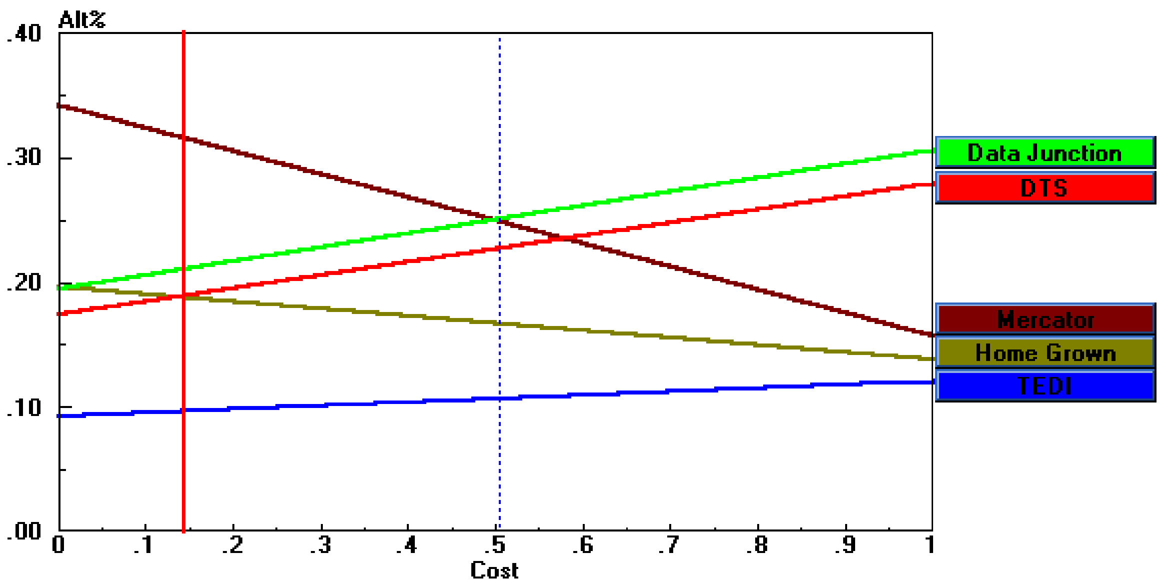

Figure 6. Gradient Analysis for Cost

$3,000, and has no transactional component to its pricing.

An interesting footnote to the analysis is that, prior to the analysis, the development staff was convinced that a homegrown approach would be the best option. The results of the decision analysis show the homegrown approach as the fourth option, contrary to intuition. Looking again at the sensitivity analysis shown in Figure 5, the homegrown option fares well in the evaluation against several objectives including usability, performance, maintenance, and feature set. Two substantive issues surfaced during the pairwise evaluations. A high level of effort would be required to provide operational support (in times of trouble there's no one else to call but Ultraprise staff…) and the overall cost in time and resources to develop and enhance the mapping capability would be prohibitive.