Data Integration: Where Does the Time Go?

Len Seligman, Arnon Rosenthal, Paul Lehner, Angela Smith
{seligman, arnie, plehner, asmith}@mitre.org

Abstract

We present a modular breakdown of data integration tasks and the results of a survey on the distribution of effort among those tasks. The modularization aids in project planning and enables portions of the work to be allocated flexibly among various human specialists, and also to automated tools. The survey results are useful for determining: (1) what are the highest value research problems to tackle? (2) where should large enterprises focus their investments in data integration tools?, and (3) how should a data integration project manager allocate people and schedule?

1 Introduction

Data integration–i.e., meaningful information exchange among systems not originally designed to work together–is a difficult challenge for large enterprises. Despite substantial progress, data integration remains expensive and labor-intensive.

In prior work, we described organizational factors–e.g., staff specialization, training costs, and participants' incentives–that are too rarely considered in data integration research. [9] proposed a repackaging of familiar data integration tasks and research problems to better fit organizational needs, describing the skills needed for each step. The goal is to move toward a more industrial process of data integration.

This paper summarizes the categorization of data integration tasks from [9] and presents the results of a survey on the distribution of effort among those tasks. Section 2 describes the task breakdown, Section 3 explains the survey procedure, and Section 4 presents its results. We close with a discussion and summary.

2 A Task Breakdown for Data Integration

The survey required a breakdown of integration tasks that was independent of any particular process or tools, instead describing products that must (at least implicitly) be part of any process. We emphasized tasks that must be performed by user organizations. Thus, we considered translation of schemas into a common data model to be out of scope, since this is increasingly provided by database and middleware vendors. The tasks are:

1. *Gather knowledge about sources* – Prior to integrating multiple data sources, one must understand the schemas, representation, and semantics of each data source.

2. *Gather knowledge about desired consumer (target) view(s)* – Similar to Task 1, but this time for the interface(s) to be used by consumer users and systems.

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

3. *Identify semantic correspondences among sources and from sources to the consumer views [6]* – In this step, one determines entities and attributes in the different systems that refer to the same (or similar) real world concepts–e.g., that Emp.Seniority in a source system can be used for Worker.YearsOfService in the consumer view.

4. *Create needed attribute transformations* – This step produces executable functions that transform attributes in the sources to properly feed the consumer views – e.g., that one must multiply HourlyWage by HoursWorked to produce Salary.

5. *Specify data combination rules* – When multiple source rows each contribute values to a single target row, how should the combination work?

   - Join or union? If a join, on what fields? Inner or outer join?
   - What result to produce when sources differ on the same fact (e.g., what is Joe's salary, really). For example, one might specify "use the mean" or "use Source1 if non-Null".

6. *Create logical mappings from sources to consumer* – Given the above information, produce an explicit mapping from sources to the target (e.g., expressed as SQL views).

7. *Data cleaning* – Discovering and correcting incorrect data values.

8. *Create and optimize an executable connection for the specific run-time environment* – e.g., a replication tool that supports an SQL subset or an extract-transform-load (ETL) tool that uses Visual Basic for transformations.

There are several advantages of this task breakdown. First, it is composed of modular, single-skill tasks. Such a breakdown aids in human resource planning, and enables portions of the work to be allocated flexibly among various human specialists (e.g., DBAs, domain experts, distributed systems programmers), and also to automated tools. Second, we believe this categorization applies to diverse integration scenarios, for example:

- Developing a data warehouse and populating it with commercial ETL tools
- XML-based information exchange among loosely coupled systems with divergent schemas
- Creating a database federation

Third, this breakdown is finer grained than previous ones. In particular, we make the knowledge capture tasks (1 and 2) first class, instead of subordinating them to others (e.g., "Schematic interschema relationship generation" and "Integrated schema generation" in the breakdown of [8]). In order to discover correspondences and discrepancies across schemas, one must learn a lot about the systems involved. We believe it is important to capture this knowledge for use in subsequent integration efforts.

Finally, the categorization identifies natural tool/expertise niches. Commercial data profilers (e.g., from Ab Initio, Ascential) support capturing source knowledge. Cupid [4] and other matchers address correspondence identification. The Context Interchange project [1] addresses creation of attribute transforms. Clio [5] supports several task categories including correspondence identification, data combination rules, producing logical mappings, and executable ones for various environments (e.g., XSL, XQuery, and SQL).

Response Type	Helpful	Not Helpful	% Helpful
All Responses (n=93)	76	17	82%
Non-MITRE (n=61)	57	4	93%
Anonymous (n=16)	14	2	88%

Table 1: Percent that found the Integration Task Categories Helpful

3 Survey Procedure

A web accessible survey was made available. Requests for participation were solicited from the following sources (the number of responses received from each is shown in parentheses):

- "All MITRE technical staff" email list, consisting primarily of systems engineers supporting U.S. government applications (33 responses)
- Referrals from MITRE technical staff, consisting of U.S. Government personnel and their contractors with data integration experience (18)
- DBWORLD email list, consisting mostly of data management researchers (23)
- DBRSRCH - A list (compiled by the authors) of selected data integration researchers (2)
- ODTUG - Oracle Development Tools Users Group email list (15)
- DB2 International Users Group email list (1)
- Comp.databases newsgroup (2)
- Participants at the Data Warehousing Institute Spring '02 World Conference (1)
- SQL Server World Users Group Development (SSWUG-DEV) email list (0)

The survey (see Appendix) contained six background questions on research or tool development experience, and whether they had experience on real integration efforts of the following types:[1] small (but nontrivial) point-to-point, small multipoint, large point-to-point, and large multipoint. Respondents were then asked to allocate percent of effort devoted to the eight categories of tasks described above. They were asked to do this for both small and large data integration efforts.

4 Results

We received 95 responses from a wide variety of organizations. Not all respondents answered all questions, but most questions had at least 75% coverage. Detailed procedures and definitions appear in the Appendix.

Normalization and Outliers. For the analyses below, the effort distribution percentages were normalized to 100. As with any questionnaire some respondents may answer randomly or dishonestly. To test for this, we excluded two "outliers" (see Appendix) and analyzed the remaining 93 responses.

Perception of Survey's Task Categorization. Overall, as indicated in Table 1, most respondents indicated that they found the task categorization described above to be helpful in thinking about data integration problems. A few respondents suggested that "integration testing" be included explicitly.

[1] In the survey, "real" meant the goal was to provide data to real users and not merely to demonstrate integration technology. "Small but nontrivial" meant 8-20 entities, while "large" was $>$ 20.

	source knowl	user view knowl	smntc crspnc	atrbt xforms	data combo rules	logical map-pings	data clng	opti-mize exctbl
Small (n=80)	17.3	11.8	13.4	12.2	10.0	11.7	14.1	9.6
Large (n=70)	18.2	11.6	14.7	11.8	11.3	9.8	14.9	7.8
Diff	(0.9)	0.2	(1.4)	0.4	(1.3)	1.9	(0.8)	1.8

Table 2: Time Allocation on Small vs Large Efforts

	source knowl	user view knowl	smntc crspnc	atrbt xforms	data combo rules	logical map-pings	data clng	opti-mize exctbl
Yes (n=55)	17.9	11.3	13.8	11.0	11.4	10.0	16.6	8.0
No (n=15)	19.4	12.5	18.1	14.6	10.6	8.9	8.7	7.3
Diff	(1.6)	(1.2)	(4.3)	(3.6)	0.8	1.1	**7.9****	0.7

Significance levels with two tailed t-test: **=p<.01 (no others attained p<.05)

Table 3: Large-scale Data Integration Experience Yes/No

Small vs. Large-scale data integration. Table 2 compares the effort estimates for small and large-scale data integration efforts (across all respondents, regardless of experience). As can be seen, there is little discrimination between the two distributions, with no statistically significant differences.

The lack of differentiation between small vs. large data integration tasks also holds true in all of the analyses we discuss below. Consequently, the rest of this section shows results only for large data integration projects. Within that set, the results are categorized by the respondent's apparent experience.

Applied data integration experience. Table 3 shows the difference between 55 respondents who claim to have large-scale data integration experience from the 15 who do not have such experience. Here there is a substantial difference on data cleaning. Specifically, experienced integrators view data cleaning as the second most effort consuming task (16.6%), while those without this experience view data cleaning as the second least effort consuming task (8.7%).

To broaden this observation, we hypothesized a correlation between data integration experience and perceived effort devoted to data cleaning. While the questionnaire did not directly ask about quantity of experience, we are able to approximate experience level by sorting respondents by:

- **2 Lrg** = Large multipoint and Large point-to-point experience
- **1 Lrg** = Either Large multipoint or Large point-to-point experience (but not both)
- **Sml MP** = Not Large experience but small multipoint experience
- **Sml PtP** = Small point-to-point experience only
- **None** = No experience at all on real integration efforts

As can be seen in Table 4, there is a clear relationship between perceived effort for data cleaning as experience increases. There also appears to be an inverse relationship between experience and estimates for specifying semantic correspondences, although the effect is not statistically significant.

Research Experience. The results shown in Table 5 reveal substantial and statistically significant differences between apparent researchers (see Appendix) and practitioners. Respondents from research sources significantly

Experience Level (see above)	source knowl	user view knowl	smntc crspnc	atrbt xforms	data combo rules	logical map-pings	**data clng**	opti-mize exctbl
2 Lrg (n=40)	14.2	10.5	13.0	11.8	11.5	11.5	**18.1**	9.3
1 Lrg (n=15)	27.6	13.4	15.9	8.8	11.2	6.1	**12.4**	4.5
Sml MP (n=9)	19.0	15.5	17.4	12.8	10.6	9.3	**9.5**	6.0
Sml PtP (n=5)	17.0	8.6	19.0	16.6	11.8	9.0	**8.0**	10.0
None (n=1)	35.0	5.0	20.0	20.0	5.0	5.0	**5.0**	5.0

Table 4: Level of Experience

	source knowl	user view knowl	smntc crspnc	atrbt xforms	data combo rules	logical map-pings	data clng	opti-mize exctbl
Research (n=16)	22.7	13.0	20.5	10.0	7.8	8.7	9.7	7.6
Other (n=54)	16.9	11.2	13.0	12.3	12.3	10.1	16.4	7.9
Diff	**5.9***	1.8	**7.4***	(2.3)	**(4.4)****	(1.4)	**(6.7)****	(0.3)

Significance levels with two tailed t-tests: **p<.01, *p<.05

Table 5: Research Sources vs Other Sources

underestimated the effort required for data cleaning and specifying data combination rules compared to respondents from other sources (p<.01). In addition, researchers overestimated the effort to gather knowledge from sources and specify semantic correspondences (p<.05).

Magnitude of Effort for Different Integration Subtasks. The bar chart in Figure 1 repeats the estimates (from Table 3) of respondents who have done large-scale data integration, and also shows standard deviations. As can be seen, the most time-consuming tasks were gathering knowledge of sources, data cleaning, and identifying semantic correspondences. The least time consuming tasks were creating logical mappings and creating and optimizing an executable connection. We discuss these results below.

5 Discussion

Retrospective estimates of time allocations on typical tasks are clearly prone to error. As shown in Figure 1, estimates for all integration task categories had high standard deviations.[2] The time estimates seemed suspiciously uniform across categories (not more than ~2:1 ratio from largest to smallest) and between estimates for large and small projects. The regularity could be an artifact of the survey procedure.

The estimates can provide a first draft for staff and cost estimates for an integration project, and ensure that none of our identified steps is forgotten. To refine the schedule, one must apply judgements about the current project, especially where estimates' variability is greatest (e.g., data cleaning). Another use is for judging progress. If one has already expended 30% of allocated project resources on (say) gathering knowledge of sources, there are problems either with the process or with the schedule.

If one averages over many projects, randomness decreases. Therefore, the estimates (if not systematically biased) can give hints to CIOs and vendors who prioritize different types of integration tools, based on potential savings of effort and of scarce categories of personnel. Data profiling tools (useful for gathering knowledge of

[2]Given the high standard deviation, measurements for only a few real projects would seem equally misleading. In addition, it was difficult to gain access to current projects.

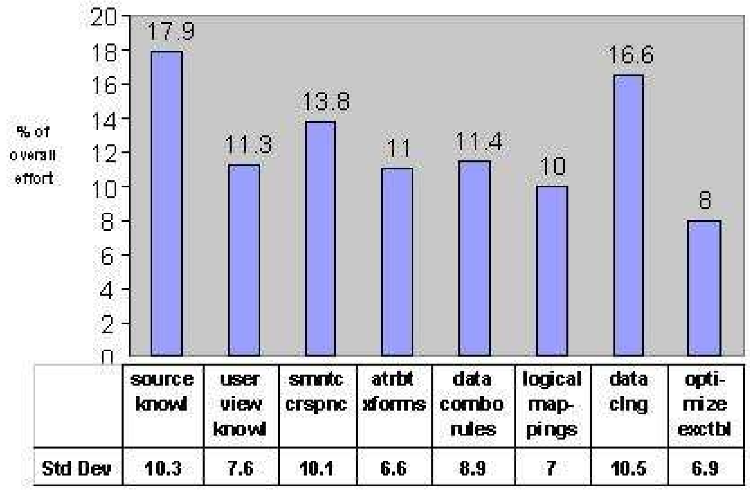

Figure 1: Percent distribution on large efforts, where respondents had done large-scale integration (n=55)

sources) and data cleaning tools (both areas where there has been commercial progress) are especially worth considering.[3]

Table 5 suggests that researchers should increase their attention to data cleaning and to specifying data combination rules. Today's tools are compendiums of simple techniques, but researchers are now seeking more power (e.g., through a combination of machine learning and human assistance) [2, 7, 10].

We were encouraged by the total number of responses received – it enabled results to be significant under 2-tailed t tests. The tactic of distinguishing users by experience was less successful. Most users claimed multiple kinds, and this was incompatible with our tactic of assuming that user estimates described the sort of project they had worked on (e.g., multi-point).

We are considering doing additional surveys. A larger response set would give larger populations for combinations of categories. However, few of our hypotheses just missed being significant, so a modest increase in sample size (e.g., by emailed requests for respondents to bring in their colleagues) seemed of marginal use. Still, a revised survey could be illuminating, with project-specific estimates tied to project characteristics, such as point-to-point vs. multipoint, types of tools used, and political environment (e.g., does your organization control the sources?). Finally, a collaboration with a contractor who does many integration efforts might give better task estimates than those obtained from respondents' memories.

The survey data is available to the community, for testing other hypotheses.

6 Summary

We adapted conventional steps of schema integration to be more friendly to project management. Each has a defined result, requires a narrow set of skills, and (in many cases) corresponds to a natural product capability.

Our survey produced interesting results, both negative and positive, though it is clearly not definitive. Most respondents (82%) found the task breakdown helpful. The reported time allocations for small and large integration problems were nearly identical. Researchers produced higher estimates for analyzing sources and for semantic correspondence than practitioners. Practitioners assigned greater time for data combination rules, and much greater weight for data cleaning; the latter effect increased with experience.

[3] In addition to data cleaning tools (which apply fixes to already bad data), managers should look seriously at improving data quality management at the sources whenever possible [3].

References

[1] C. Goh, S. Bressan, S. Madnick, M. Siegel. Context Interchange: New Features and Formalisms for the Intelligent Integration of Information. TOIS, 17(3), 1999

[2] M. Hernndez, S. Stolfo. The Merge/Purge Problem for Large Databases. SIGMOD Conference 1995

[3] D. Loshin. Enterprise Knowledge Management: The Data Quality Approach, Morgan Kaufmann, 2001

[4] J. Madhavan, P. Bernstein, E. Rahm. Generic Schema Matching with Cupid. VLDB 2001

[5] R.J. Miller, M. Hernndez, L. Haas, L. Yan, H. Ho, R. Fagin, L. Popa. The Clio Project: Managing Heterogeneity. SIGMOD Record, 30(1), 2001

[6] E. Rahm, P. Bernstein: A survey of approaches to automatic schema matching. VLDB Journal 10(4)

[7] V. Raman, J. Hellerstein. Potter's Wheel: An Interactive Data Cleaning System. VLDB 2001

[8] S. Ram, V. Ramesh. Schema Integration: Past, Present, and Future, in A. Elmagarmid, M. Rusinkiewicz, A. Sheth, Management of Heterogeneous and Autonomous Database Systems, Morgan Kaufmann, 1999

[9] A. Rosenthal, L. Seligman, S. Renner, F. Manola. Data Integration Needs an Industrial Revolution. International Workshop on Foundations of Models for Information Integration (FMII-2001), Viterbo, Italy, September 2001, http://www.mitre.org/resources/centers/it/staffpages/arnie/pubs/Foundations01.pdf

[10] L.-L. Yan, R.J. Miller, L. Haas, R. Fagin. Data-Driven Understanding and Refinement of Schema Mappings. SIGMOD Conf., 2001.

Appendix: Methodology Details

This section provides details and rationales for our analysis, to help make the effort repeatable. The survey text and the response data are at http://www.mitre.org/resources/centers/it/staffpages/arnie/SurveyPaper-DataEng

Definitions

***"Researchers"*:** Self-rating of research experience was of little use. 70% of respondents claimed research experience, including respondents from sources consisting primarily of practitioners (e.g., 53% of the ODTUG respondents). From this, we concluded that interpretations varied considerably on what it means to "perform research in data integration"– probably from researching alternate products for a warehouse implementation, to performing original research aimed at peer-reviewed conferences or journals. In the future, we suggest that surveys make the latter definition explicit.

Given this problem, we instead distinguished researcher vs. nonresearcher by comparing the two primarily research-oriented sources of respondents (i.e., dbworld and "dbrsrch," our list of selected data integration researchers) against responses from the other sources, which consisted primarily of practitioners. This gave statistically significant results, though self-identification did not.

Outliers: If a respondent provided a task percentage that was more than five standard deviations away from the mean response, then that respondent's data was dropped from the analysis. Two respondents were rejected using this test. Upon inspection, one of these was an obvious coding error. Perhaps the other had an idiosyncratic interpretations of the definitions.

Procedures and Rationales

We wanted a short questionnaire, since longer ones get fewer responses. We felt that we could ask for estimates of only two situations, and opted for long versus short projects. Beyond this, we used self-described experience or source of the response to differentiate kinds of efforts.

For the multipoint/point-to-point dichotomy, there was substantial overlap in the populations. Of those that gave estimates for large projects, 49 indicated they had done both multipoint and point-to-point, 13 said they had done only multipoint, while only 7 said they had just done point-to-point. There were no substantial or statistically significant differences between those who had only done point-to-point vs. those who had only done multipoint.

Since large and small gave similar behavior, and we were especially interested in large-scale integration problems, most of the data presented was for large-scale efforts.

The hardcopy questionnaire indicated that responses would be normalized to 100, while the Web version indicated that the total should equal 100 and displayed a running total as respondents completed the questionnaire. Of the 95 responses, only 4 used the hardcopy version.

While participant selection was not random, we have no reason to believe that self-selected participation was based on reasons other than curiosity and a willingness to help.