CHAPTER 5: RetroGuide Evaluation

Vojtech Huser

5.1 Introduction

This chapter presents a RetroGuide (RG) evaluation study which was conducted to assess the flowchart-based modeling approach. This study complements the previously presented case studies in an overall effort to evaluate the RG project.

The evaluation was targeted at informatics users with small to moderate analytical experience. Due to RG's key developmental goal to lower the technological barriers for novice users to analyze data stored in an Enterprise Data Warehouse (EDW), this specific target group was deemed most appropriate for the evaluation.

A literature review was conducted, looking at how projects similar in nature to RG were evaluated and what study designs and potential measures were used. This review is described in section 5.2. The rest of the chapter describes the design, methodology, results, and discussion of the RG evaluation.

5.2 Literature review

Friedman [1] defines several types of evaluation studies which are performed depending on the maturity of the evaluated resource. The list of those types, organized by different stages of development of the resource, would be: (1) needs assessment, (2) design validation, (3) structure validation, (4) usability test, (5) laboratory function study, (6) field function study, (7) laboratory user effect study, (8) field user effect study, and (9) problem impact study.

Many of the reviewed projects which (similarly to RG) try to offer a new analytical methodology or technology were evaluated only in a prototype stage, because many projects never reached a stage where the technology would be refined to a user-friendly final product. The evaluation would rarely reach Friedman's study levels of field function or field user effect. Moreover, many of the products would mainly be used within the originating institutions, and the developers' goals would not even include development of a complete analytical product which could be installed at other institutions. In a new domain, where most of the projects were conducted in an academic setting, this is not surprising. All these considerations apply also to the RG project and its evaluation.

Dorda et al. [2, 3] at the University of Vienna evaluated their internal ArchiMed analytical system, which uses a custom-developed language called AMAS. In terms of evaluation, their articles present only ArchiMed's usage statistics at a single hospital or hospital department to demonstrate the usefulness of their system. They did not quantitatively compare ArchiMed or AMAS language capabilities to any other widely-available analytical technology, e.g., Structured Query Language (SQL). The evaluation section of their key article [2] focuses, for the most part, on "lessons learned," and merely enumerates the challenges and key issues encountered during the development and subsequent use of their system.

Nigrin et al. [4, 5] at Harvard University evaluated DXTractor. It is an internally-developed and internally-used system for composing temporal queries utilizing a sophisticated graphical user interface (GUI). Their key article [4] mostly presents DXTractor's design issues and theoretical problems in temporal querying, and lacks any rigorous evaluation results. The article provides seven clinical query examples together with how those examples would be implemented in DXTractor's internal, set-based query syntax. They do not present usage statistics, and they admit that they did not perform any formal evaluation or comparison to other analytical packages. In the discussion section, they mention a very informal evaluation using four clinicians from their institution. Each clinician was asked to compose five very basic queries. The complexity of the queries they used in this small-scale evaluation is, however, very low compared with the overall DXTractor capabilities. They also admit in the discussion that more sophisticated temporal queries may be awkward to run routinely by nonexpert users because of their length and complexity.

Das and his team at Stanford University evaluated their Time Line SQL language (TLSQL) used by their internal temporal mediator system called Chronus [6]. Their article, similarly to those by the above mentioned groups, mainly focuses on highlighting extensions of the TLSQL language. It presents four clinical query examples with the corresponding TLSQL code. Unlike the previous examples, however, they did perform a very small quantitative evaluation where they compared query execution time using TLSQL versus standard SQL. For both technologies, they present query time in seconds for three queries against a database of 60 patients with a total of 301 clinic visits. The execution time of all three evaluated queries was longer with TLSQL. Despite this slower execution time with TLSQL, another study done by the same team [7] states that certain queries which are possible in TLSQL cannot be performed in SQL.

Finally, the ACT/DB system developed by Nadkarni's team is a notable exception in terms of use at other institutions. The TrialDB system, which is based on ACT/DB, is available as an open-source application and is used by several institutions worldwide [8]. The reason might be that later enhancements of their system focused on emphasizing support for clinical trial data management (e.g., integrating data collected at multisite trials [9]), and the goal to support advanced query functionally became only secondary. Although two publications point to interesting advanced query capabilities [10, 11], there is no published evaluation study which would formally assess the capabilities of the Query Kernel of ACT/DB or compare it to other analytical technologies or packages.

In conclusion, the reviewed evaluation designs did not suggest any optimal study methodology for evaluating the RG system. This review confirmed that picking an optimal design and quantitative measures for evaluation of new analytical technologies is not trivial. Like the reviewed studies, RG is in the prototype stages of development, and is used internally at one institution. On Friedman's scale, the selected evaluation goal was at the third, resource structure validation level.

Related to this argument of initial structure validation is the nature of the overall RG project as a feasibility test of workflow technology in the medical domain. The overall objective was to evaluate whether the idea of using a workflow-based approach is feasible and useful for solving medical, analytical, retrospective problems. The goal was to evaluate the methodological approach as opposed to the current user interface of the

JaWE open source workflow editor, which was selected because it represented the best open-source workflow editor at the time of this project.

5.3 Methods

Several important design decisions had to be made prior to deciding on the final evaluation study design. The first decision was whether to do a stand-alone evaluation looking only at RG, or a comparative evaluation where RG would be compared with existing technology. The comparative approach was ultimately chosen since results demonstrating beneficial differences against an established technology would be more persuasive for current analysts and requestors. As for the chosen comparison standard, SQL-based analytical technology was selected, which is similar to Das' study. The advantage of the choice of SQL is that it represents an established standard and is not proprietary to any specific package or vendor. Full availability of the comparison technology to the evaluating team, considering also the software price, was also a limiting factor in the choice of the comparison standard.

The second consideration was whether to evaluate the RG suite as a whole or only focus on certain parts of the framework. This consideration is related to the prototype nature of many of the components of the RG suite and the overall focus on the feasibility of use of workflow technology. The two key possible areas were: (1) input: how a user in a given technology models the analytical problem for execution, i.e., the form in which the question has to be asked; and (2) output: how useful is the output generated by the technology, i.e., the form in which the answers are obtained. The chosen final evaluation design focused on the first aspect because a number of possible future improvements are possible with the RG generated output, whereas RG's question modeling paradigm dictated by a workflow-technology-based approach will not fundamentally change in any future versions.

The third consideration was the enrollment criteria for the participants and realistic expectations on how a large sample can be obtained for different pools of users. RG scenario modeling has two major categories of users: (1) clinicians who can use RG to review scenarios and in some advanced clinical-user cases even partially author them; and (2) analysts who are best qualified to fully author the scenario's code layer. Moreover, participant time requirements were probably the most important limiting factor. A minimum introduction into RG technology, taking at least 30-45 minutes, was unavoidable in almost all possible designs, and then the actual study time, depending on the complexity of the content, was also substantial. Friedman, in his evaluation textbook [1], defines a category of *proxy users* of an evaluated resource, and that category is employed in settings where the circumstances prevent enrolling enough resource-intended users. The final study enrollment criteria were that subjects had both experience with analytical problems, databases and SQL, and a biomedical informatics background. The second criterion, which limited the subjects to existing or past biomedical informatics students, tried to ensure that the enrolled subjects would have some aspects of a clinical-user type and at the same time some aspects of an analyst-user type.

The final evaluation study had two main aims: The first aim (quantitative) was to look at the ability of subjects to solve analytical tasks using particular methods (SQL vs. RG), as well as understand and modify existing solutions. The second aim (qualitative) was to look at the user's experience with RG-technology compared to SQL.

The study was operationalized as a paper study packet where participants were asked to answer a series of questions. Corresponding to the two aims, the study had two parts. The first, quantitative part focused on measuring performance on a test with a series of questions. The second, qualitative part included open-ended and other types of questions designed to compare RG and SQL in terms of user experience and future intention to use flowchart-based analytical technology.

5.3.1 Quantitative evaluation

Two sets of questions were used in the quantitative evaluation. The first set had nine task questions (T1-T9), and the second set had five multiple choice questions (C1-C5). All participants were asked to answer the set of 14 questions twice – once using SQL and once using RG. The participants were randomized so that half of them started with RG while the other half started with SQL. Participants were instructed to have a 24-hour interval between the two approaches tested (wash-out period).

Each of the nine task questions (T1-T9) involved an analytical task to be solved, and a large blank space was left below the questions for the participant to provide, in the SQL case, a verbal, pseudo-code solution, or, in the RG case, a drawing of a flowchart solution annotated with the employed RetroGuide external applications (RGEA). For example, question T4 was: "Find all patients who had at least 2 creatinine lab results flagged as too high"; question T6 was: "Find all patients who have diabetes but no record of hypertension diagnosis." Appendix B lists all study task questions. Figure 5.1 shows an example of one participant's RG solution to question T4.

The subjects were instructed to focus on the overall ability to solve a given task and describe fundamental steps necessary for solving a given task. Minor technology syntax errors were specifically stated as outside the focus of the task questions. Each task question would be later scored 0, 0.5, or 1 point. A correct solution received 1 point, a partial solution containing at least 50% of the necessary steps received 0.5 points, and an incorrect or no solution received 0 points. For scoring purposes, each task question had, apart from several general query correctness criteria (for RG and SQL), an itemized set of crucial solution steps. The presence of those crucial steps in a participant's solution was evaluated on a percentage range (0-100%) which was then translated into test points. For example, the list for RG's solution to question T5 ("Find all patients who had at least 2 creatinine lab result too high but they must be at least 180 days (or more) apart from each other") included: (1) proper identification of two, abnormal creatinine results; and (2) correct implementation of the 180-days-apart criterion. Partial fulfillment of those crucial

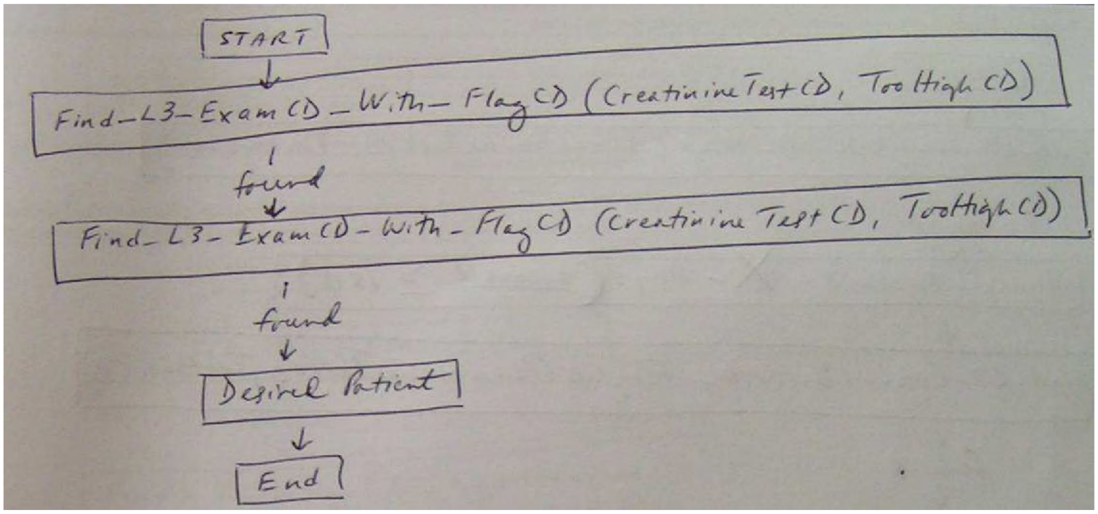

START
Find-L3-Exam()-With-Flag() (CreatinineTest(), TooHigh())
found
Find-L3-Exam()-With-Flag() (Creatinine Test(), TooHigh())
found
Desired Patient
End

Figure 0.1 Sample participant's RG solution to task question T4

T4 question was: Find all patients who had at least 2 creatinine lab results flagged as too high. This solution was fully correct and was scored 1 point. The concept of current EHR position and the forward parsing strategy ensures that the two laboratory event steps will capture two distinct creatinine lab results.

steps was also considered; however the final, three-grade scoring granularity (0, 0.5, or 1 point) for each question simplified the solution scoring. Two reviewers were used to assign scores for each task question. The reviewers were senior-level informaticists, each with over 12 years of experience in medical data analysis and decision support.

Participants were instructed to use no more than 45 minutes to solve all task questions in a given methodology (RG or SQL). The use of the time limit ensured that the study also considered the time required for solving a problem in a given methodology. The analytical approach, which is faster for participants to use, had a higher chance of achieving higher average test scores.

The five choice questions (C1-C5) introduced a solution to a given analytical problem1 and then presented a slightly extended problem2 together with three possible solutions (marked a, b, or c). The participants were asked to choose the correct option. The c option in each question was "none of the above" and was designed to be the correct answer on one of the five questions for each tested approach to broaden the number of options [12]. Choice questions were later scored 1 point for a correct solution and 0 points for an incorrect or missing one. All five choice questions for RG and SQL, respectively, are listed in Appendix B.

A total test score for RG answers and for SQL answers was calculated. A paired t-test was used to compare the mean RG and SQL scores. The null, two-sided hypothesis was that there is no difference in mean scores of RG versus SQL technology. Each of the two subsections (choice questions and task questions) was also analyzed separately using the same statistical approach. The task questions were evaluated in three subanalyses: separately for each reviewer and as a combined average score.

The first purpose for the quantitative evaluation was to statistically compare the two analytical methods. The second purpose was to expose the participant to a set of real analytical problems in both technologies, and to use this practical exposure as a basis for the subsequent qualitative study. Although the set of analytical problems used in the study was meant to be realistic and was reviewed by a panel of experts (researchers with experience and formal training in informatics and research), it does not cover all possible analytical tasks, and it was not realistic to attempt to do so. The choice of tasks, their total number, and their complexity was, in fact, greatly limited by the overall study time requirements and by the clinical and analytical background knowledge of the study participants ultimately targeted for enrollment (current or past biomedical informatics students). Furthermore, in terms of task selection there was a special focus of the overall evaluation on tasks which include temporal conditions, since temporal considerations are frequent and important in medicine and current technologies lack sufficient support for them [2].

An initial, larger set of 21 possible problems (for task and choice questions) was reduced to only 14 final questions using a panel of two study researchers. The smaller set was then presented to five medical informatics experts. Two experts provided suggestions which were later incorporated into the final set.

5.3.2 Qualitative evaluation

The second part of the study focused on qualitatively comparing the two tested approaches (SQL vs. RG) using 13 follow-up questions (F1-F13) divided into two major sections A and B. Section A focused on both compared technologies or a generic analytical technology. Section B questions were targeted only at the RG technology.

The first three questions (F1-F3) of section A were open- ended questions. Question F1 asked what approach was preferred by the participant and why. Questions F2 and F3 asked the participant to state all disadvantages found with SQL and RG, respectively. Section A concluded with one question (F4) about the quality of the instructions provided for both parts of the quantitative study, and a final question (F5) where 9 features of a generic analytical tool were ranked according to the subject's perception of importance.

Section B used Likert scale questions (F6-F13) which were based on previously validated constructs and survey instruments examining the acceptance of technologies. Technology acceptance models enable prediction of use of a given technology within an organization by measuring several predicting factors. It enables assessment of actual later use of a new technology by using only a sample of users.

The key utilized model was the Unified Theory of Acceptance and Use of Technology (UTAUT) published by Venkatesh et al. [13]. UTAUT unifies several previous technology acceptance models. In biomedical informatics, the Technology Acceptance Model (TAM) described by David and Bagozzi [14, 15], one of the UTAUT predecessors, has been used in several informatics technology evaluations [16-19]. UTAUT defines four direct determinants (performance expectancy, effort expectancy, social influence, and facilitating conditions) of user *intention to use* a technology and later *actual use* of the technology. It also identifies four moderating factors which affect the above four determinants (gender, age, experience, and voluntariness of use). UTAUT clearly identifies what factors play important roles in acceptance of new technology. This is highly relevant to the RG evaluation, aimed at demonstrating the benefit of RG as a new technology.

Section B questions investigated 3 UTAUT constructs by using at least 2 questions per construct: performance expectancy (questions F6, F9, F12), effort expectancy (questions F7, F10, F13), and behavioral intention (questions F8, F11). UTAUT provides a validated 5-point Likert scale ranging from 1 (=strongly disagree) to 5 (=strongly agree). For example, question F6 investigating performance expectancy was "I find RetroGuide useful for solving analytical problems"; question F10 investigating effort expectancy was "It is easy for me to use RetroGuide to create analytical models"; and question F8 investigating behavioral intention was "Having RetroGuide as an available option in my analytical job – I intend to use RetroGuide." Knowledge of the user's assessment of those three constructs for RG technology, according to the UTAUT model, enables prediction of later actual use of RG-like technology. All qualitative questions are listed in Appendix B.

Knowledge of the average Likert scores for the three investigated UTAUT parameters enables some prediction of the final UTAUT outcome parameter of *actual use*. This prediction is, however, limited by the fact that none of the qualitative evaluation's questions attempted to assess UTAUT's *social influence* and *facilitating conditions* predictors. Given the structure validation setting of the study and the prototype nature of the tested resource, it did not make sense to include these two predictors.

5.3.3 Background questionnaire

The initial part of the study packet included a background questionnaire which consisted of five questions. The questions were based on the moderators defined by the UTAUT (age, gender, past education, level of SQL experience) and the subject's source of SQL knowledge (formal SQL course or self-learning). All background questions are listed in Appendix B.

5.3.4 Study procedure

The subjects were enrolled using email and a personal 5-minute interview describing the goals and structure of the study. Enrolled subjects met with the investigator for 45 minutes where standard training on RG technology was presented and they were given the actual study packet. The packet also contained 3 additional items: (1) summary of the RG technology training; (2) a simplified list of necessary RG external applications; and (3) sample data sheet demonstrating the storage structure of the EHR data which applied to both compared technologies (SQL and RG). Training for SQL technology was not provided since at least basic knowledge of SQL was an enrollment criterion for participants, and the study assumed that a basic knowledge is more than equivalent to the 45-minute RG training. The study packet contained specific instructions for the study participant to perform the study on his/her own schedule, including the 24-hour break between the compared approaches and the time-limit restrictions for each section.

After return of all study packets, the qualitative section was then scored and analyzed using R statistical package [20]. Manual content analysis of the qualitative part was done using categorization, summarization and tabulation techniques on the collected textual data [21]. The reliability of the tested UTAUT constructs in the qualitative section B was analyzed using the same statistical package.

5.4 Results

A total of 19 subjects were enrolled into the study. One subject later dropped out and did not return the study packet so that the analyzed sample size was 18 subjects. Table 5.1 shows subject characteristics: gender, age, SQL experience, the source of SQL knowledge, and educational background. The majority of the participants were female (72.2%) and of age category 31-35 (33.3%). Intermediate knowledge of SQL was most prevalent (44.4%), with no participants with no or expert SQL knowledge, which were the desired enrollment criteria.

5.4.1 Quantitative evaluation

The mean total RG score was 11.1 compared with the mean SQL score of 6.3. The mean difference of 4.8 ± 1.8 was statistically significant using paired t-test ($p<<0.001$, 95% CI 3.4 - 5.4). Similarly significant results were found when looking at subscores of task questions (T1-T9) only, as well as subscores of choice questions (C1-C5) only. See Table 5.2 for complete results. Task questions T1-T9 were scored by two different reviewers. The total score results presented above and in the Table 5.2 use the average score from both reviewers. A separate analysis of the scores for each reviewer on the task analysis is shown in Table 5.3.

The Cohen's kappa statistic was used to determine the degree of agreement of the two reviewers on the total pool of 234 task question score pairs. The result (kappa=0.53) indicates moderate agreement [22]. A subanalysis by task-type (SQL vs. RG) showed a difference in agreement on SQL scores (kappa=0.66) versus RG scores (kappa=0.23). The lower agreement on RG scores can be explained by differences in scoring a repeated error between the two reviewers: Reviewer A took off points for the first occurrence of

Table 0.1 Evaluation study subject characteristics

Property	**Number (%)**
N	18
Sex	
Male	5 (27.8%)
Female	13 (72.2%)
Age	
18-25	1 (5.6%)
26-30	0 (0 %)
31-35	6 (33.3%)
35-40	5 (27.8%)
41-45	1 (5.6%)
45-50	4 (22.2%)
50+	1 (5.6%)
Database experience	
none	0 (0%)
extremely basic	6 (33.3%)
basic	4 (22.2%)
intermediate	8 (44.4%)
expert	0 (0%)
Source of expertise*	
medical informatics db class	12 (66.6%)
db class elsewhere	3 (16.6%)
self-learning	12 (66.6%)
Educational background*	
computer science degree	0 (0%)
MD degree	8 (44.4%)
RN degree	5 (27.8%)
MS in medical informatics	4 (22.2%)
PhD in medical informatics	5 (27.8%)
MS in nursing informatics	5 (27.8%)
PhD in nursing informatics	1 (5.6%)
MS in other field	5 (27.8%)
PhD in other field	2 (11.1%)

* multiple choice is possible for source of expertise background question; the total will thus not equal n.

Table 0.2 Quantitative evaluation results: mean scores, standard deviations, and paired t-test confidence intervals and p-values

	mean	**SD**	**CI; p value ***
Total score (range 0 – 14)			
SQL total score	6.3	2.2	
RG total score	11.1	1.9	
difference RG-SQL	4.8	1.8	3.4 - 5.4; p<<0.001
Task questions (range 0 - 9)**			
SQL	4.3	1.6	
RG	7.3	1.2	
difference RG-SQL	3.0	1.3	2.4 - 3.6; p<<0.001
Choice questions (range 0 - 5)			
SQL	1.9	1.2	
RG	3.2	1.1	
difference RG-SQL	1.3	1.3	0.7 - 2.0; p= 0.0004

* 95% confidence interval, p-values are for paired t-test, 2 sided hypothesis
** mean scores (from both reviewers) are shown for task questions (T1-T9)

Table 0.3 Qualitative evaluation: Task questions results analyzed separately for each reviewer

	mean	**SD**	**CI; p value ***
Task questions (Reviewer A)			
SQL	4.5	1.6	
RG	7.8	1.2	
difference RG-SQL	3.3	1.4	3.1 - 3.7; p<<0.001
Task questions (Reviewer B)			
SQL	4.0	1.6	
RG	6.7	1.3	
difference RG-SQL	2.7	1.4	2.0 - 3.3; p<<0.001

The range for all items was 0-9 points

an error but not for later repeats of the same error, while reviewer B took off points for each occurrence. The two reviewers still agreed on 65% of the task scores.

Linear regression (LR) was used to determine whether score difference could be predicted by any of the participant characteristics such as gender, age, SQL experience, or SQL experience source. No LR model could predict the score difference (adjusted R-squared < 0.1) and none of the factors were statistically significant.

A two-sample t-test showed no statistical difference in test score differences between the group which started with the SQL approach versus the group which started with the RG approach.

5.4.2 Qualitative evaluation

In question F1 ("Which technology do you prefer and why?"), 94.4% (17 subjects) preferred RG to SQL. Analysis of the qualitative comments was performed and several categories were identified. The leading categories were "easy to learn/use/understand" (9 subjects), followed by "temporal modeling capabilities" (6 subjects), and "more intuitive/natural/logical" (4 subjects). One subject found both methodologies equivalent. No subjects preferred SQL. Such strong preference for RG, in fact, represents probably the most important overall result of the evaluation study.

In the question on SQL's difficulties (question F2), the leading category of comments were "must know exact syntax/be expert" (7 subjects), "difficult to use" (6 subjects), and "insufficient support for temporal criteria" (5 subjects). The leading categories for RG's difficulties (question F3) were "need to know function of various applications and new terminology" (4 subjects), "none" (3 subjects), "hard to understand what data user gets back" (2 subjects), and "can be slow for queries involving a larger population" (2 subjects).

For two questions which asked about clarity of study instructions, using a 5-point scale of very poor (1) to excellent (5), the SQL mean result was 4.06 and the RG mean was 4.24. Those scores indicate that the study instructions for both methods were sufficiently clear. Finally, for question F5 where the importance of several features of a generic analytical tool was evaluated, the results are shown in Table 5.4. The table lists overall rank for 9 prelisted features and one participant-added feature. The two most important features were "intuitive model understood by nonexperts" (rank 2.22) and "short training time" (rank 3.61). These were followed by five other factors with similar average rank, ranging from 5.05 to 5.56 ("graphical representation," "facilitation of collaboration," "short query time," "based on established technology," and "direct access to data as physically stored").

For section B, where 3 UTAUT constructs were investigated (questions F6-F13), Table 5.5 lists Likert scale means and standard deviations for each question. It also shows Cronbach's alpha reliability statistics for each concept. All three surveyed constructs of performance expectancy (PE), effort expectancy (EE), and behavior intention (BI) exhibit reliability within the recommended range of greater then 0.70 [23]. Employing the UTAUT prediction model, the achieved mean scores indicate that if the RG technology would be developed from the current prototype into a full software product, it most likely would be well accepted by future users (all scores are well above the middle neutral score of 3, which indicates a favorable effect on the final measure of actual technology use).

Table 0.4 Average ranking of features of a generic analytical tool

Overall rank *	Description
2.22	modeling paradigm is intuitive even for a nonexpert
3.61	training time required to master the tool is short
5.06	graphical representation of the problem
5.06	presence of features which facilitate collaboration of an informaticist/clinician with a professional database analyst
5.22	query time is short
5.35	tools incorporate an established technology standard or syntax
5.56	technology offers direct access data as they are physically stored in the data warehouse
6.06	technology supports iterative working cycle of a project team (extending previous results and analyses)
8.22	affordable purchase price for the software
n/a	other: support for time based structures

* Smaller number indicates higher importance. Only one participant entered an additional free-text feature (exact rank calculation was not possible)

Table 0.5 Construct reliability, mean and standard deviation scores (5-point Likert scale)

Construct	mean	SD	Cronbach's alpha
Performance expectancy (PE)			0.871
PE1	4.45	0.62	
PE2	4.11	0.58	
PE3	3.89	0.76	
Effort Expectancy (EE)			0.849
EE1	4.39	0.70	
EE2	4.28	0.75	
EE3	3.89	0.76	
Behavior intention (BI)			0.752
BI1	4.22	0.88	
BI2	4.39	0.70	

5.4.3 Conclusion

This resource structure validation study compared RG with SQL-based tools using a sample of nonexpert users. Using the RG approach, the subjects achieved significantly higher scores in solving analytical tasks, and also scored higher in tasks which required understanding of given analytical solutions. The study demonstrated that most users preferred RG to SQL because RG was easier to learn, it better supported temporal tasks, and it seemed to be a more logical modeling paradigm. Using UTAUT technology acceptance prediction model, the study results suggest that a fully developed, RG-like technology likely would be well accepted by users.

This evaluation study has several limitations. The fist limitation is its focus on users with low to intermediate levels of SQL knowledge. Thus the results are valid only for this particular group of users. However, one of the aims of the RG project was to offer a tool which would lower the barrier to analysis of data in an EDW for novice users (novice in terms of analytical expertise and understanding the warehouse structure). Expert users of SQL do not need support in solving tasks which they can solve already using their existing knowledge. Thus, given the RG targeted user group of RG, this limitation of the study is acceptable.

The second, perhaps more serious, limitation is the choice of tasks used in the quantitative part of the study. Although the final set of questions was reviewed by a panel of informatics experts, the bias towards selecting tasks more suitable for a particular approach is possible, and a special focus on tasks including temporal tasks is acknowledged.

However, in light of the first and the second limitation, it is important to restate that the additional purpose (other than the statistical score analysis) of the task and choice questions was to provide a practical exposure to the two compared technologies and prepare the participants for the subsequent qualitative part of the study, which is much less affected by these two limitations. While reviewing the limitations, it also is equally important to note that this was a prototype, structure validation study, and that there is little consensus or clear guidance on how structure validation studies of prototype resources should be rigorously conducted. The literature review in section 5.2 demonstrated the limited number of publications where data on quantitative evaluation design and measures were included in articles describing alternative analytical technologies. Friedman [1] describes the structure validation category of studies only very briefly and literally concludes that "anything useful" can be used as a potential measure. In his enumeration of potential measures, however, he includes tests and test score measures as valid outcome variables.

There are two evaluation studies in the biomedical informatics domain which have very similar evaluation design and also use a task-based approach and score comparison [24, 25]. For example, Campbell did a study which compared five bedside information resources [24]. Her sample size (16 subjects) was similar to the RG evaluation and her compared test scores ranged from 0 to 5 points. These two studies give some confirmation that the employed task-based design and score comparison has a precedent for small scale, resource evaluation studies.

5.5 References

[1] C. P. Friedman and J. Wyatt, *Evaluation methods in biomedical informatics*, 2nd ed. ed: New York : Springer, c2006, 2006.

[2] W. Dorda, W. Gall, and G. Duftschmid, "Clinical data retrieval: 25 years of temporal query management at the University of Vienna Medical School," *Methods Inf Med*, vol. 41, pp. 89-97, 2002

[3] W. Gall, P. Sachs, G. Duftschmid, and W. Dorda, "A retrieval system for the selection and statistical analysis of clinical data," *Med Inform Internet Med*, vol. 24, pp. 201-12, 1999

[4] D. J. Nigrin and I. S. Kohane, "Temporal expressiveness in querying a time-stamp--based clinical database," *J Am Med Inform Assoc*, vol. 7, pp. 152-63, 2000

[5] D. J. Nigrin and I. S. Kohane, "Data mining by clinicians," *Proc AMIA Symp*, pp. 957-61, 1998

[6] A. K. Das and M. A. Musen, "A temporal query system for protocol-directed decision support," *Methods Inf Med*, vol. 33, pp. 358-70, 1994

[7] A. K. Das and M. A. Musen, "A comparison of the temporal expressiveness of three database query methods," *Proc Annu Symp Comput Appl Med Care*, pp. 331-7, 1995

[8] What is TrialDB?, http://ycmi.med.yale.edu/trialdb/trialdb_contents.htm#What%20is%20TrialDB? [accesed: December 10, 2006]

[9] C. A. Brandt, P. Nadkarni, L. Marenco, B. T. Karras, C. Lu, L. Schacter, J. M. Fisk, and P. L. Miller, "Reengineering a database for clinical trials management: lessons for system architects," *Control Clin Trials*, vol. 21, pp. 440-61, 2000

[10] P. M. Nadkarni and C. Brandt, "Data extraction and ad hoc query of an entity-attribute-value database," *J Am Med Inform Assoc*, vol. 5, pp. 511-27, 1998

[11] A. M. Deshpande, C. Brandt, and P. M. Nadkarni, "Temporal query of attribute-value patient data: utilizing the constraints of clinical studies," *Int J Med Inform*, vol. 70, pp. 59-77, 2003

[12] Good Practice Guide in Question and Test Design, http://www.pass-it.org.uk/resources/031112-goodpracticeguide-hw.pdf [accesed: October 14, 2006]

[13] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, "User acceptance of information technology: Toward a unified view.," *MIS Quarterly*, vol. 27, pp. 425-478, 2003

[14] F. D. Davis, "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology.," *MIS Quarterly*, vol. 3, pp. 319-40, 1989

[15] R. P. Bagozzi, F. D. Davis, and P. R. Warshaw, "Development and Test of a Theory of Technological Learning and Usage," *Human Relations*, vol. 45, pp. 659-86, 1992

[16] N. C. Hulse, G. Del Fiol, and R. A. Rocha, "Modeling end-users' acceptance of a knowledge authoring tool," *Methods Inf Med*, vol. 45, pp. 528-35, 2006

[17] W. G. Chismar and S. Wiley-Patton, "Test of the technology acceptance model for the internet in pediatrics," *Proc AMIA Symp*, pp. 155-9, 2002

[18] E. V. Wilson and N. K. Lankton, "Modeling patients' acceptance of provider-delivered e-health," *J Am Med Inform Assoc*, vol. 11, pp. 241-8, 2004

[19] P. J. Hu, "Examining the Technology Acceptance Model Using Physician Acceptance of Telemedicine Technology," *Journal of Management of Information Systems*, vol. 16, pp. 91-112, 1999

[20] R-Team, *R: A language and environment for statistical computing.* Vienna, Austria: R Foundation for Statistical Computing, 2006.

[21] A. Lacey and D. DLuff, "Qualitative research and data analysis," *University of Nottingham, UK (Trent Focus Group)*, 2004

[22] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, pp. 159-174, 1977

[23] J. C. Nunally, *Pyschometric Theory. 2nd ed.* New York: McGraw-Hill, 1978.

[24] R. Campbell and J. Ash, "Comparing bedside information tools: a user-cente task-oriented approach," *AMIA Annu Symp Proc*, pp. 101-5, 2005

[25] W. R. Hersh, "A task-oriented approach to information retrieval evaluation," *Journal of the American Society for Information Science*, 1996

Appendix B

EVALUATION STUDY

SECTION 1: Background questionnaire questions

B1: What is your educational background? (check one or multiple boxes)

□ CS (computer science)
□ MD (medicine)
□ RN (nursing)
□ MS – Biomedical/Medical Informatics (pursuing or completed)
□ PhD – Biomedical/Medical Informatics (pursuing or completed)
□ MS – Nursing Informatics (pursuing or completed)
□ PhD – Nursing Informatics (pursuing or completed)
□ MS – Other domain (pursuing or completed)
□ PhD – Other domain (pursuing or completed)

B2: How would you classify your SQL experience? (check only one answer)

□ none	*no knowledge of SQL commands*
□ extremely basic	*passive knowledge of the following basic constructs: (SELECT a,b,c FROM table WHERE xxx)*
□ basic	*experience with active use of basic constructs: (SELECT a,b,c FROM table WHERE xxx)*
□ intermediate	*knowledge of most constructs offered by SQL (including JOIN, GROUP BY, MIN, MAX, COUNT)*
□ expert	*knowledge and extensive experience with writing advanced SQL queries (20+ lines of command code)*

B3: What is the source of your SQL experience? (multiple answers are OK)

□ database class at informatics department (which year:________________________)

□ database class during my previous education (which year:______________________)

□ self-learning education during previous project (which year:_____________________)

B4: What is your gender? (check only one answer)

□ male □ female

B5: What is your age group? (check only one answer)

□ 18-25 □ 26-30 □ 31-35 □ 36-40 □ 41-45 □ 46-50 □ 50+

SECTION 2: Quantitative study: Task questions

T1: EYE EXAM REPORT

Find all patients who ever had in their EHR any record of a previous "Comprehensive Eye Exam" report.

Hint:

"Comprehensive Eye Exam Report" is a case of textual report. It would be documented as event subtype code under level 2. Level 1 code to event type would be "Clinical Text Data".

T2: CREATININE TEST

Find all patients who had blood creatinine level measured at least once.

Hint:

"Creatinine, serum/plasma, quantitative" is a case of laboratory test. It would be documented as event L1_type ("Standard Lab Data"), L2_subtype ("Chemistry 7 panel") and L3_exam ("Creatinine, serum/plasma quantitative").

T3: CREATININE TEST FLAGGED AS TOO HIGH

Find all patients who ever had blood creatinine lab result which was flagged as being too high.

Hint:

The database record of such test would be the same as in previous question with some extra information. Creatinine tests with too high value would have extra code in the flag column. Field FLG_CD would contain code for too high result (TooHighCD).

T4: CREATININE TOO HIGH AT LEAST TWICE

Find all patients who had at least 2 creatinine lab results flagged as too high.

Hint:

Question is related to the previous two questions T3 and T2 and how data would be stored in the database was shown above. Suppose that all items in EVENTS table underwent a quality assurance process and there are no erroneous duplicate items.

T5: CREATININE TOO HIGH TWICE AND AT LEAST 180 DAYS APART

Find all patients who had at least 2 creatinine lab result too high but they must be at least 180 days (or more) apart from each other.

Hint:

Question is again related to previous three questions and requires additional temporal criterion.

Focus only on creatinine lab results flagged as too high. Consider any normal (unflagged) creatinine lab results irrelevant for this task.

T6: DIABETIC BUT NOT HYPERTONIC

Find all patients who have diabetes but no record of hypertension diagnosis.

Hint:

ICD billing codes will be used to identify diabetes and hypertension. Patient is considered diabetic if "ICDdiabetesCD" is found at least once in his record. Similarly patient is considered hypertonic if "ICDhypertensionCD" is found at least once in his record. For simplicity, assume that only single code is used for diabetes and single code is used for hypertension. Those codes, however, may be present in the record multiple times. After a patient has been assigned a diagnosis of diabetes or hypertension, each separately billed hospital visit will have those codes re-entered.

T7: BOTH DIABETIC AND HYPERTONIC

Find all patients who have record of both diabetes and hypertension diagnosis in their EHR. The temporal order does not matter.

Hint:

Question is related to the previous questions T7. Use the same hint and the database sample record which was shown there.

T8: FIRST DIABETIC, THEN HYPERTONIC

Find all patients who were *first* diagnosed with *diabetes* and their diagnosis of *hypertension* came *after* their diabetes (certain temporal order of conditions enforced).

Hint:

Question is related to the previous questions T6 and T7. Same sample data apply as shown in question T6.

Remember that there can be multiple billing entries for diabetes and hypertension. Time of disease onset is important for this question. Consider onset time as the time of the first and earliest appearance of the diabetes or hypertension billing code in the EHR. Patients where diabetes and hypertension started on the same date is not considered as “after” and such patients should not be listed.

T9: LONG-TERM SYSTOLIC BLOOD PRESSURE CONTROL IN DIABETIC PATIENTS

Find all patients with diabetes, who have at least 2 systolic blood pressure measurements over 130 mmHg after they became diabetic. However, 2 additional restrictions apply to the question. First, do not consider any blood pressure measurement in the initial treatment period of 24 months after establishing the diagnosis of diabetes. Second, the 2 elevated systolic blood pressure measurements over 130 must be at least 11 months apart from each other.

Hint:
Systolic blood pressure value is an observation type of event. It would be documented as event L1_type (“Observation Event”), L2_subtype (“Vital Signs Panel”) and L3_exam (“Blood Pressure, Systolic.”).

SECTION 3: Quantitative study: Choice questions

PART A) SQL

C1: LDL CHOLESTEROL TEST

INTRODUCTION:

You are analyzing patients with cardiovascular conditions.
Consider the following problem to solve.
Problem 1: Find all patients who ever had LDL-cholesterol over 130 mg/dl.
Look at the provided correct solution
Solution to problem 1:

```
SELECT DISTINCT PT_ID
FROM EVENTS
WHERE L3_EXAM_CD = 'LDLcholesterol_CD'
AND VAL_NUM > 130
```

QUESTION:

Solve related problem 2:
Problem 2: *Find all patients whose latest LDL cholesterol in year 2005 was over 130 mg/dl.*
Several solutions are provided bellow. Circle one correct answer.

A)

```
SELECT DISTINCT PT_ID
FROM
 (SELECT PT_ID, max(EV_TIME), VAL_NUM
 FROM EVENTS
 WHERE L3_EXAM_CD = 'LDLcholesterolCD'
 AND EV_TIME <= '2005-DEC-31' AND EV_TIME >= '2005-JAN-01'
 GROUP BY PT_ID)
WHERE VAL_NUM > 130
```

B)

```
SELECT DISTINCT a.PT_ID
FROM
 (SELECT PT_ID, max(EV_TIME), L3_EXAM_CD
 FROM EVENTS
 WHERE L3_EXAM_CD = 'LDLcholesterolCD'
 AND EV_TIME <= '2005-DEC-31' AND EV_TIME >= '2005-JAN-01'
 GROUP BY PT_ID, L3_EXAM_CD) a,
EVENTS b
WHERE a.PT_ID = b.PT_ID AND a.EV_TIME = b.EV_TIME AND a.L3_EXAM_CD = b.L3_EXAM_CD
AND b.VAL_NUM > 130
```

C) none of the above

C2: FRACTURE IN WOMEN

INTRODUCTION:

You are analyzing the problem of osteoporosis in female patients.
Consider the following problem to solve.
Problem 1: Find all patients who had a fracture.
Look at the provided correct solution. FractureCDs is enumeration of all ICD codes for fractures.
Solution to problem 1:

```
SELECT DISTINCT PT_ID
FROM EVENTS
WHERE L1_TYPE_CD = 'ICDDiagnosisCD' AND TERM2_CD = 'FractureCDs'
```

QUESTION:

Solve related problem 2:
Problem 2: *Find all patients who had a fracture at age 66.*
Several solutions are provided bellow. Circle one correct answer.
Hint: *Every EHR record starts with <Birth Event> which shows the day of birth.*

A)

```
SELECT DISTINCT a.PT_ID
FROM
 (SELECT PT_ID,EV_TIME
 FROM EVENTS
 WHERE L1_TYPE_CD =  'BirthEventCD' ) a,
 (SELECT PT_ID, EV_TIME
 FROM EVENTS
 WHERE L1_TYPE_CD = 'ICDDiagnosisCD' AND TERM2_CD IN ('FractureCDs')) b,
WHERE a.PT_ID = b.PT_ID
AND (b.EV_TIME – a.EV_TIME)/365.25 >= 66
AND (b.EV_TIME – a.EV_TIME)/365.25 < 67
```

B)

```
SELECT DISTINCT a.PT_ID
FROM
 (SELECT PT_ID,EV_TIME
 FROM EVENTS
 WHERE L1_TYPE_CD =  'BirthEventCD' ) a,
 (SELECT PT_ID, min(EV_TIME)
 FROM EVENTS
 WHERE L1_TYPE_CD = 'ICDDiagnosisCD' AND TERM2_CD IN ('FractureCDs')
 GROUP BY PT_ID) b,
WHERE a.PT_ID = b.PT_ID
AND (b.EV_TIME – a.EV_TIME)/365.25 >= 66
AND (b.EV_TIME – a.EV_TIME)/365.25 < 67
```

C) none of the above

C3: HYPERTENSION PRIOR DIABETES

INTRODUCTION: You have a group of patients visiting your clinic and you are analyzing the relationship of diabetes and hypertension.
Consider the following problem.
Problem 1: Find all patients who have both conditions – diabetes and also hypertension. (temporal order does not matter)
Look at the provided solution.
Solution to problem 1:

```
(SELECT DISTINCT PT_ID
FROM EVENTS
WHERE TERM2_CD = 'DiabetesCD' AND L1_CD_TYPE = 'ICDDiagnosisCD'
)
INTERSECT
(SELECT DISTINCT PT_ID
FROM EVENTS
WHERE TERM2_CD = 'HypertensionCD' AND L1_CD_TYPE = 'ICDDiagnosisCD'
)
```

QUESTION:*Solve related problem 2:*
Problem 2: *Find patients who have both conditions but they had diagnosis of hypertension first and after that became diabetic (specific order enforced).*
Several solutions are provided bellow. Circle one correct answer.

A)

```
SELECT DISTINCT a.PT_ID
FROM
(SELECT PT_ID, min(EV_TIME) as DIAB_ONSET
FROM EVENTS
WHERE TERM2_CD = 'DiabetesCD' AND L1_CD_TYPE = 'ICDDiagnosisCD'
GROUP BY PT_ID) a,
(SELECT PT_ID, min(EV_TIME) as HYPER_ONSET
FROM EVENTS
WHERE TERM2_CD = 'HypertensionCD' AND L1_CD_TYPE = 'ICDDiagnosisCD'
GROUP BY PT_ID) b
WHERE a.PT_ID = b.PT_ID
and (b.HYPER_ONSET - a.DIAB_ONSET) < 0
```

B)

```
SELECT DISTINCT PT_ID
FROM
(
(SELECT PT_ID, EV_TIME as DIAB_TIME
FROM EVENTS
WHERE TERM2_CD = 'DiabetesCD' AND L1_CD_TYPE = 'ICDDiagnosisCD'
)
INTERSECT
(SELECT PT_ID, EV_TIME as HYPER_TIME
FROM EVENTS
WHERE TERM2_CD = 'HypertensionCD' AND L1_CD_TYPE = 'ICDDiagnosisCD'
)
) WHERE  (HYPER_TIME - DIAB_TIME) < 0
```

C) none of the above

C4: COUNT NUMBER OF FRACTURE EPISODES

INTRODUCTION:

You are analyzing a group of female patients over 80 years-old for the total number of fractures. The best method to identify fractures is using ICD billing codes. Consider the following problem.

Problem 1: Count how many fracture episodes each patient had.
(using a predefined set of ICD codes for fractures referred to as *'FractureCDs'*.
Look at the provided solution.

Solution to problem 1:

```
SELECT PT_ID, count(*) as total
FROM EVENTS
WHERE TERM2_CD  in ('FractureCDs') AND L1_CD_TYPE = 'ICDDiagnosisCD'
GROUP BY PT_ID
```

In this solution to problem 1, a fracture with 3 follow-up visits will produce 4 result rows. In an expanded task, you would like to approximate to the number of actual fractures a patient experienced and not simply count all fracture-related visits. So in expanded problem, you would like to exclude fracture follow-up visits. Most fractures are resolved within 90 days.

QUESTION:

Solve related problem 2:

Problem 2: *Count the fracture episodes each patient had. But after any given fracture episode, do not count any follow-up fracture visit within 90 days.*
Several solutions are provided bellow. Circle one correct answer.

A)

```
SELECT PT_ID, count(*) FROM
 (SELECT PT_ID, EV_TIME
 FROM EVENTS
 WHERE TERM2_CD  in ('FractureCDs') AND L1_CD_TYPE = 'ICDDiagnosisCD'
 GROUP BY PT_ID) a
FULL OUTER JOIN
(SELECT PT_ID, EV_TIME
 FROM EVENTS
 WHERE TERM2_CD  in ('FractureCDs') AND L1_CD_TYPE = 'ICDDiagnosisCD'
WHERE a.PT_ID  = b.PT_ID
HAVING a.EV_TIME – b.EV_TIME > 90
GROUP BY PT_ID
```

B)

```
SELECT PT_ID,
  count(*) OVER (PARTITION BY (EV_TIME,90) ORDER BY PT_ID) as total
FROM EVENTS
WHERE TERM2_CD  in ('FractureCDs') AND L1_CD_TYPE = 'ICDDiagnosisCD'
GROUP BY PT_ID
```

C) None of the above.

C5: ADVERSE DRUG EVENT DETECTION

INTRODUCTION: You are investigating the problem of respiratory Adverse Drug Event (ADE) after the use of narcotics.

Consider the following problem.

Problem 1: Find all patients who were given narcotic antidote naloxone and within 6 hours from this naloxone administration were transferred to ICU. If naloxone is given multiple times, consider only the first such episode. Look at the provided solution.

Solution to problem 1:

Hint: Inpatient administered drugs are recorded as level 4 code under "Medication Event" type; Transfer to ICU event has level 4 coded value 'toICUTransferCD' under 'Tranfer Event' type

```
SELECT DISTINCT PT_ID
FROM EVENTS tran
 JOIN
 (SELECT PT_ID, min(EV_TIME) as EV_TIME
 FROM EVENTS
 WHERE L4_CD_CODED_VALUE= 'NaloxoneCD' AND L1_CD_TYPE = 'MedicationEventCD'
 GROUP BY PT_ID) nlx
ON tran.PT_ID=nlx.PT_ID
WHERE tran.L1_CD_TYPE =  'TransferEventCD' and tran.L4_CD_CODED_VALUE = 'toICUTransferCD'
and  tran.EV_TIME - nlx.EV_TIME BETWEEN 0 and 0.25  /* 6 hours = 0.25 days*/
```

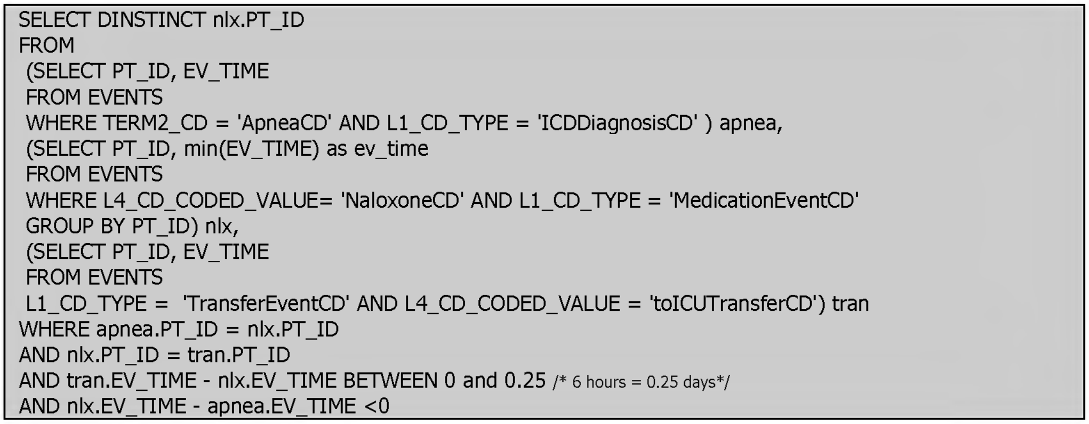
QUESTION: ***Solve related problem 2:***

Problem 2: *Find all patients who experienced the above ADE (naloxone and transfer within 6 hours) and also had a record of sleep apnea ICD diagnosis prior to this ADE. (i.e.: the searched sequence is: apnea, naloxone and transfer) Several solutions are provided bellow. Circle one correct answer.*

A)

```
SELECT DINSTINCT nlx.PT_ID
FROM
 (SELECT PT_ID, EV_TIME
 FROM EVENTS
 WHERE TERM2_CD = 'ApneaCD' AND L1_CD_TYPE = 'ICDDiagnosisCD' ) apnea,
 EVENTS tran
   JOIN
 (SELECT PT_ID, min(EV_TIME) as ev_time
 FROM EVENTS
 WHERE L4_CD_CODED_VALUE= 'NaloxoneCD' AND L1_CD_TYPE = 'MedicationEventCD'
 GROUP BY PT_ID) nlx
 ON tran.PT_ID = nlx.PT_ID
WHERE tran.L1_CD_TYPE =  'TransferEventCD' AND tran.L4_CD_CODED_VALUE = 'toICUTransferCD'
AND nlx.PT_ID = tran.PT_ID
AND tran.EV_TIME - nlx.EV_TIME BETWEEN 0 and 0.25 /* 6 hours = 0.25 days*/
AND nlx.EV_TIME - apnea.EV_TIME <0
```

B)



```
SELECT DINSTINCT nlx.PT_ID
FROM
 (SELECT PT_ID, EV_TIME
 FROM EVENTS
 WHERE TERM2_CD = 'ApneaCD' AND L1_CD_TYPE = 'ICDDiagnosisCD' ) apnea,
 (SELECT PT_ID, min(EV_TIME) as ev_time
 FROM EVENTS
 WHERE L4_CD_CODED_VALUE= 'NaloxoneCD' AND L1_CD_TYPE = 'MedicationEventCD'
 GROUP BY PT_ID) nlx,
 (SELECT PT_ID, EV_TIME
 FROM EVENTS
 L1_CD_TYPE =  'TransferEventCD' AND L4_CD_CODED_VALUE = 'toICUTransferCD') tran
WHERE apnea.PT_ID = nlx.PT_ID
AND nlx.PT_ID = tran.PT_ID
AND tran.EV_TIME - nlx.EV_TIME BETWEEN 0 and 0.25 /* 6 hours = 0.25 days*/
AND nlx.EV_TIME - apnea.EV_TIME <0
```

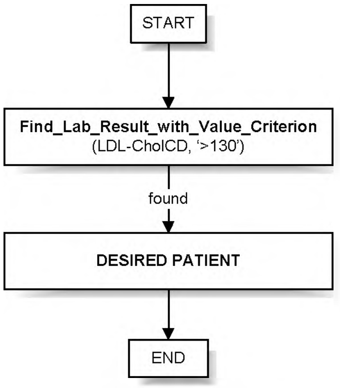
C) none of the above

<u>PART B) RetroGuide</u>

C1: LDL CHOLESTEROL TEST

INTRODUCTION:

You are analyzing patients with cardiovascular conditions.
Consider the following problem to solve.
Problem 1: Find patients who ever had LDL-cholesterol over 130 mg/dl.
Look at the provided correct solution
Solution to problem 1:

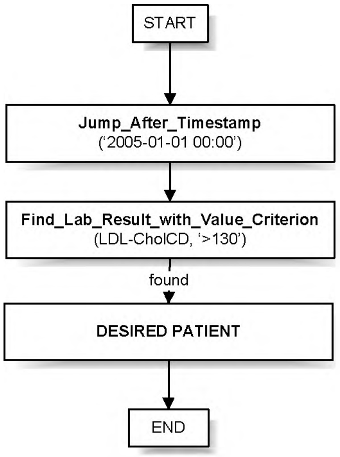


QUESTION:

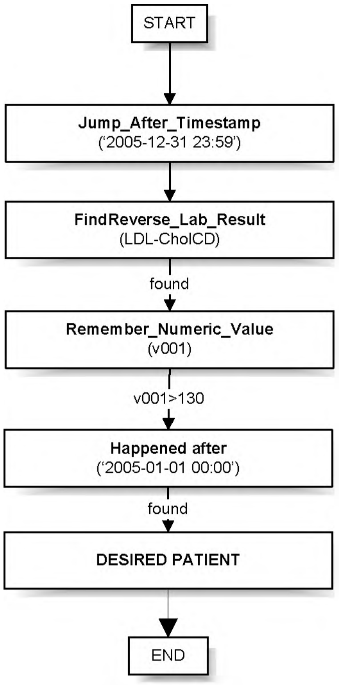
Solve related problem 2:

Problem 2: *Find all patients whose latest LDL cholesterol in year 2005 was over 130 mg/dl.*
Several solutions are provided bellow. Circle one correct answer.

A)



B)

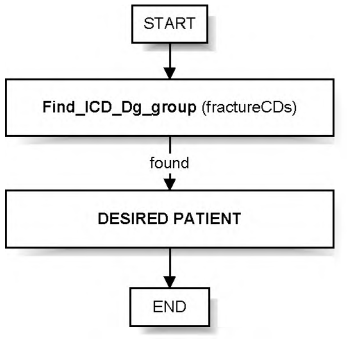


C) none of the above

C2: FRACTURE IN WOMEN (RG)

INTRODUCTION:

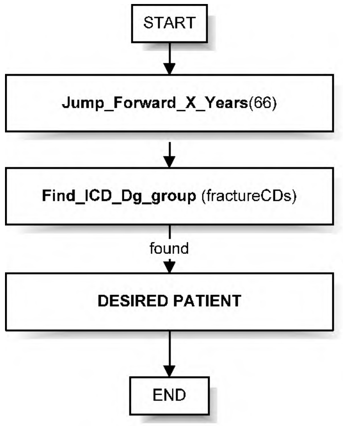
You are analyzing the problem of osteoporosis in female patients.
Consider the following problem to solve.
Problem 1: Find all patients who had a fracture.
Look at the provided correct solution. FractureCDs is enumeration of all ICD codes for fractures.
Solution to problem 1:



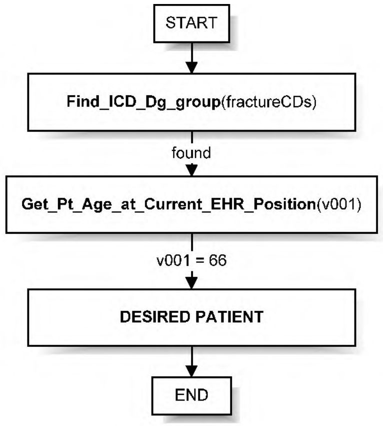
QUESTION:

Solve related problem 2:
Problem 2: *Find all patients who had a fracture at age 66.*
Several solutions are provided bellow. Circle one correct answer.
Hint: Every EHR record starts with <Birth Event> which shows the day of birth.

A)



B)


START
Find_ICD_Dg_group(fractureCDs)
found
Get_Pt_Age_at_Current_EHR_Position(v001)
v001 = 66
DESIRED PATIENT
END

C) none of the above

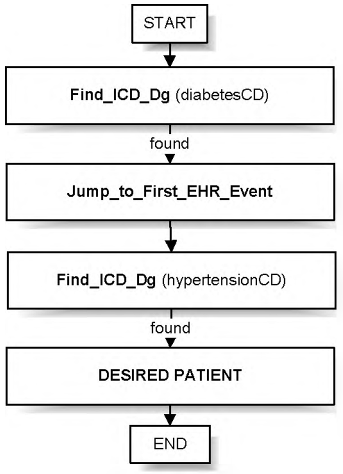
C3: HYPERTENSION PRIOR DIABETES

INTRODUCTION: You have a group of patients visiting your clinic and you are analyzing the relationship of diabetes and hypertension.

Consider the following problem

Problem 1: Find all patients who have both conditions – diabetes and also hypertension. (temporal order does not matter) Look at the provided solution.

Solution to problem 1:

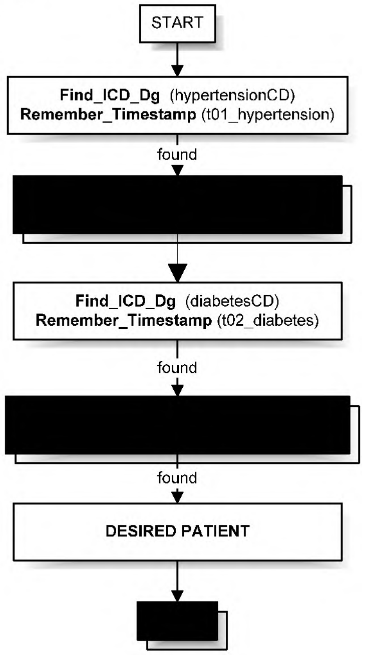


QUESTION: ***Solve related problem 2:***

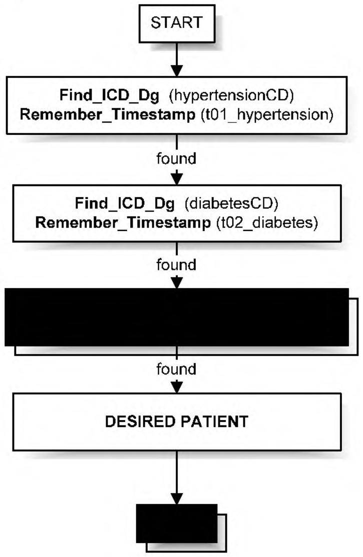
Problem 2: *Find all patients who have both conditions but they had diagnosis of hypertension first and after that became diabetic (specific order enforced).*

Several solutions are provided bellow. Circle one correct answer.

A)


START
Find_ICD_Dg (hypertensionCD)
Remember_Timestamp (t01_hypertension)
found
[redacted]
Find_ICD_Dg (diabetesCD)
Remember_Timestamp (t02_diabetes)
found
[redacted]
found
DESIRED PATIENT
[redacted]

B)


START
Find_ICD_Dg (hypertensionCD)
Remember_Timestamp (t01_hypertension)
found
Find_ICD_Dg (diabetesCD)
Remember_Timestamp (t02_diabetes)
found
[redacted]
found
DESIRED PATIENT
[redacted]

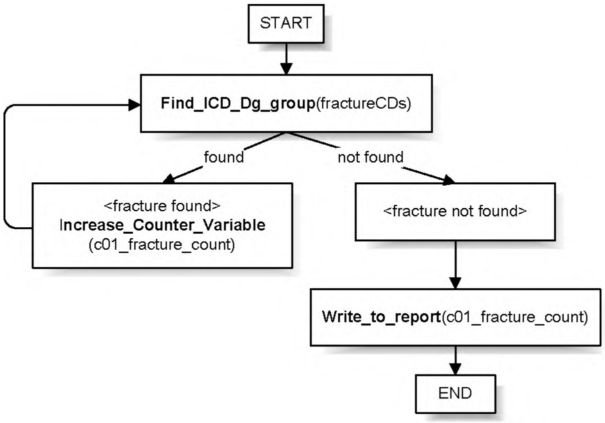
C) none of the above

C4: COUNT NUMBER OF FRACTURE EPISODES

INTRODUCTION: You are analyzing a group of female patients over 80 years-old for the total number of fractures. The best method to identify fractures is using ICD billing codes. Consider the following problem.

Problem 1: Count how many fracture episodes each patient had. (using a predefined set of ICD codes for fractures referred to as *'FractureCDs'*.

Solution to problem 1:


START
Find_ICD_Dg_group(fractureCDs)
found
not found
<fracture found>
Increase_Counter_Variable
(c01_fracture_count)
<fracture not found>
Write_to_report(c01_fracture_count)
END

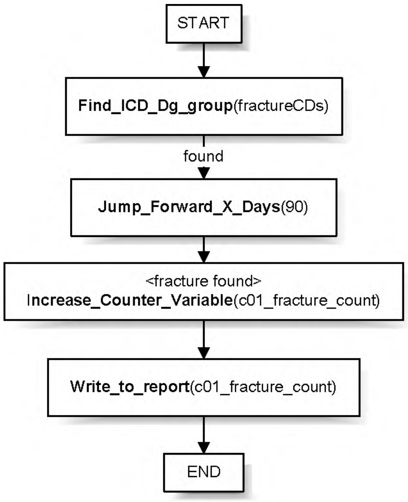
In this solution to problem 1, a fracture with 3 follow-up visits will produce 4 result rows. In an expanded task, you would like to approximate to the number of actual fractures a patient experienced and not simply count all fracture-related visits. So in expanded problem, you would like to exclude fracture follow-up visits. Most fractures are resolved within 90 days.

QUESTION: ***Solve related problem 2:***

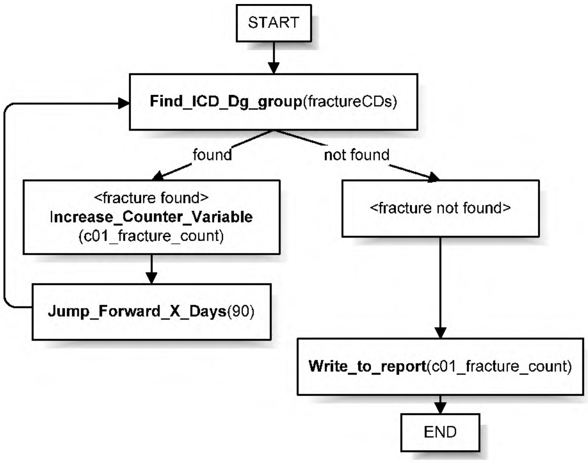
Problem 2: *Count the fracture episodes each patient had. But after any given fracture episode, do not count any follow-up fracture visit within 90 days.*

Several solutions are provided bellow. Circle one correct answer.

A)


START
Find_ICD_Dg_group(fractureCDs)
found
Jump_Forward_X_Days(90)
<fracture found>
Increase_Counter_Variable(c01_fracture_count)
Write_to_report(c01_fracture_count)
END

B)



C) none of the above

C5: ADVERSE DRUG EVENT DETECTION

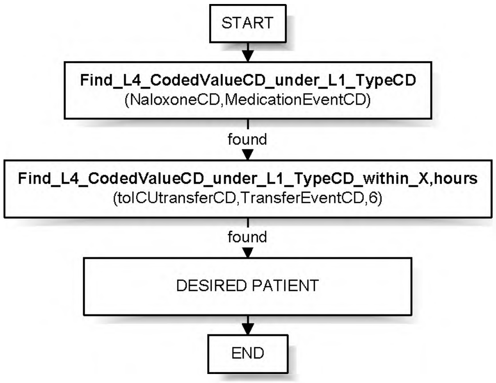
INTRODUCTION:

You are investigating the problem of respiratory Adverse Drug Event (ADE) after the use of narcotics. Consider the following problem.

Problem 1: Find all patients who were given narcotic antidote naloxone and within 6 hours from this naloxone administration were transferred to ICU. If naloxone is given multiple times, consider only the first such episode. Look at the provided solution.

Solution to problem 1:

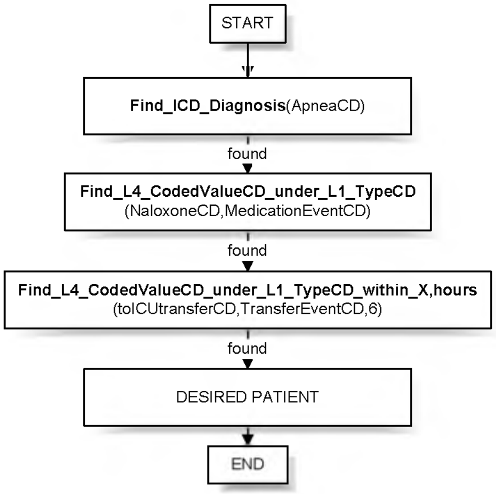
Hint:Inpatient administered drugs are recorded as level 4 code under "Medication Event" type;Transfer to ICU event has level 4 coded value 'toICUTransferCD' under 'Tranfer Event' type


START
Find_L4_CodedValueCD_under_L1_TypeCD
(NaloxoneCD,MedicationEventCD)
found
Find_L4_CodedValueCD_under_L1_TypeCD_within_X,hours
(toICUtransferCD,TransferEventCD,6)
found
DESIRED PATIENT
END

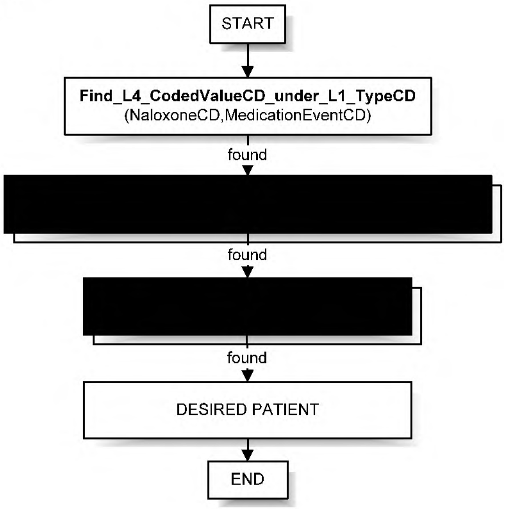
QUESTION:*Solve related problem 2:*

Problem 2: *Find all patients who experienced the above ADE (naloxone and transfer within 6 hours) and also had a record of sleep apnea ICD diagnosis prior to this ADE. Several solutions are provided bellow. Circle one correct answer.*

A)


START
Find_ICD_Diagnosis(ApneaCD)
found
Find_L4_CodedValueCD_under_L1_TypeCD
(NaloxoneCD,MedicationEventCD)
found
Find_L4_CodedValueCD_under_L1_TypeCD_within_X,hours
(toICUtransferCD,TransferEventCD,6)
found
DESIRED PATIENT
END

B)


START
Find_L4_CodedValueCD_under_L1_TypeCD
(NaloxoneCD,MedicationEventCD)
found
[redacted]
found
[redacted]
found
DESIRED PATIENT
END

C) none of the above

SECTION 4: Qualitative study questions

PART A) COMPARISON AND GENERAL QUESTIONS:

The following 4 questions are about the two approaches you used in the test.

F1: Which approach did you prefer for solving and understanding/modifying tasks (SQL-based tools vs. RetroGuide)
(check one option)

□ SQL-based tools
□ both approaches are equivalent (no preference)
□ RetroGuide

and provide a short explanation for you choice (why?) (e.g., what features make the difference): (free text question)

F2: What difficulties or disadvantages you see in using SQL-based approach for solving or modifying analytical tasks? (free text question)

F3: What difficulties or disadvantages you see in using RetroGuide approach for solving or modifying analytical tasks? (free text question)

F4a: Was the introductory information for SQL-based-tools part of the test sufficient? (e.g., database schema, sample data) (check one option)

□very poor □poor □fair □good □excellent

F4b: Was the introductory information for RetroGuide part of the test sufficient? (e.g., flowchart modeling approach, list of available external applications) (check one option)

□very poor □poor □fair □good □excellent

The final question in this section is a general question about a generic analytical tool.

F5: Please check features which you consider important for any modeling and analytical tool in general.
Also assign rank 1-10, next to checked features, according to their importance to you (1=most important, 10 = least important).

rank

□ ___ graphical representation of the problem

□ ___ presence of features which facilitate collaboration of an informaticist/clinician with a professional database analyst

□ ___ tools incorporate an established technology standard or syntax

□ ___ training time required to master the tool is short

□ ___ modeling paradigm is intuitive even for a nonexpert

□ ___ query time is short

□ ___ affordable purchase price for the software

□ ___ technology supports iterative working cycle of a project team (extending previous results and analyses)

□ ___ technology offers direct access data as they are physically stored in the data warehouse

□ ___ other: please state___________________________________

PART B) RETROGUIDE SPECIFIC QUESTIONS:

This section contains questions only about RetroGuide.

For questions in this section – consider the same scenario which was used in the test: Suppose again, that you are a clinical analyst for an imaginary FutureDecade Healthcare. Your company just purchased a tool exactly like RetroGuide for all analysts in the company.

Circle a number on a scale from 1(=strongly disagree) to 5 (=strongly agree) as your answer to the following set of questions.

F6: I find RetroGuide useful for solving analytical problems/questions.

(strongly disagree) 1 2 3 4 5 (strongly agree)

F7: I find RetroGuide easy to use.

(strongly disagree) 1 2 3 4 5 (strongly agree)

F8: Having RetroGuide as an available option in my analytical job - I intend to use RetroGuide.

(strongly disagree) 1 2 3 4 5 (strongly agree)

F9: Using RetroGuide enables me to accomplish analytical tasks more quickly.

(strongly disagree) 1 2 3 4 5 (strongly agree)

F10: It is easy for me to use RetroGuide to create analytical models.

(strongly disagree) 1 2 3 4 5 (strongly agree)

F11: Having RetroGuide as an available option in my analytical job - I predict that I would use RetroGuide.

(strongly disagree) 1 2 3 4 5 (strongly agree)

F12: Using RetroGuide increases my productivity.

(strongly disagree) 1 2 3 4 5 (strongly agree)

F13: Learning to use RetroGuide is easy for me.

(strongly disagree) 1 2 3 4 5 (strongly agree)

PhD dissertation title, abstract

ANALYZING BIOMEDICAL DATA SETS USING EXECUTABLE GRAPHICAL MODELS

by
Vojtech Huser

ABSTRACT

Clinical data warehouses accumulate large amounts of terminology-coded data. In addition to increased accumulation of data, higher data granularity, and longer time-spans, there is also an increasing demand for analysis of this data. For a nonexpert, the ability to analyze this data unaided is very limited. To address this problem, I developed an analytical framework that works with flowchart models which can be extended with modular external applications and executed on retrospective data. This framework was inspired by emerging workflow technology. Workflow technology offers several tools which support modeling, execution, and extensive analysis of IT or organizational processes. The three specific aims of this dissertation were to review workflow technology and its current use, develop an analytical framework which utilizes graphical, process-based modeling, called RetroGuide (RG), and evaluate this framework using a series of case studies and a formal, comparison evaluation study.

RG's graphical representation format facilitates a stepwise, procedural approach to formulating analytical tasks. It uses a single patient execution model, and it resembles a manual chart review methodology. RG models can model complex temporal conditions and utilize external data manipulation, statistical, or reasoning technologies. The representation format is split into two layers, a flowchart and a code layer, which improves collaboration of analytical team members. Reports generated automatically by RG allow advanced drill-down capabilities, show in detail the model's execution trail for each analyzed patient, and support iterative model improvements.

Within this dissertation, three analytical domains of quality improvement, decision support development, and medical research were explored. Seven case studies which utilize the Enterprise Data Warehouse (EDW) at Intermountain Healthcare are described (e.g., quality improvement problems in osteoporosis and cardiovascular patients, analysis of a computerized glucose management protocol, a problem in adverse drug event monitoring, or a research analysis of cancer patients). These case studies demonstrate RG's ability to support a wide range of complex analytical tasks, facilitate iterative exploration and review of electronic health record data, and provide a testing

environment for retrospective simulation of analytical or decision support processes (using data from a real, large EDW).

Finally, a formal comparison study involving modeling analytical tasks in RG and Structured Query Language (SQL), and a qualitative study of RG are presented. The results suggest that RG's modeling approach is intuitive and easy to use, enables better modeling of the evaluated set of analytical tasks, and is preferred over SQL by a group of nonexpert data analysts.