Software Industry

The following guidelines are intended to provide examples of experimental development projects which would qualify for Canadian SR&ED (Scientific Research & Experimental Development) tax credits.

Content Summary:

1100 – Software SRED – General Guidelines

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
Performance (step response, throughput) (seconds)	5	2	Yes
CPU load (% busy)	90	60	Yes
Stability (# Errors/ run)	20	1	No
Foot Print (Mb)	10	2	Yes

[NOTE: THESE GUIDELINES ARE REPRODUCED FROM EXERPTS OF, "GUIDANCE ON ELIGIBILITY OF SOFTWARE PROJECTS FOR THE SR&ED TAX CREDITS," AS PUBLISHED BY THE CRA IN CO-OPERATION WITH CATA & THE SOFTWARE INDUSTRY, SEPTEMBER 2000.]

Advancement - Note that an advancement in technology can rarely be described by listing software functions and features at an "end-user" level. Advances are typically made through innovation in software architectures, designs, algorithms, techniques or constructs within the field of information technology or computer science. The advancement need not be large.

Note: Simply claiming to have developed the first or best software suite for a given purpose does not in itself prove that the taxpayer has made a technological advancement. A new and unique software suite can be built using only well known combinations of constructs, tools and methods without technological advancement. This is analogous to designing and building a unique and complex office building without making any advancements in the field of civil engineering.

Evidence of Technological Advancement could include credible third party literature or comparisons of the capabilities sought against those previously available from the taxpayer himself. As in a court of law, there are no rigid definitions of what constitutes credible evidence.

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Internet searches	12 sites / articles	We searched extensively and found no out of the box solutions to meet our criteria
Patent searches	43 patents	We looked at 43 patents that were simlar to our requirements but were insufficient
Competitive products or processes	3 products	We looked at 3 other companies working with similar technologies
Potential components	7 products	There are seven potential components we are looking at using
Queries to experts	5 responses	We spoke to Derek Tarko P. eng He was helpful but advised us this wasn't available

Hint: As a means to identify the advancement(s), the taxpayer might identify the technological reason why his architecture or technique was not used before. How does it compare with earlier solutions or with the current solution of a competitor? What earlier technical constraint has been overcome?

Field of Science/Technology:

Software engineering and technology (2.02.09)

Project Details:

Intended Results:	Develop new processes, Develop new materials, devices, or products, Improve existing processes, Improve existing materials, devices, or products
Work locations:	Commercial Facility
Key Employees:	Bill Gates (Computers - B.Sc. (1981) / CEO), Steve Jobs (Software - B.Eng. (1983) / Project

Manager), Sergery Brin (Unknown / Unknown)

Evidence types: Records of resources allocated to the project, time sheets; Design of experiments; Project records, laboratory notebooks; Design, system architecture and source code; Records of trial runs; Progress reports, minutes of project meetings; Test protocols, test data, analysis of test results, conclusions; Photographs and videos; Samples, prototypes, scrap or other artefacts

Scientific or Technological Advancement:

Uncertainty #1: Define variables for experimentation (OPTIMAL METHODS)

The objective here is to outline options for developing sets of questions which may act as catalyst to provide an effective and efficient method of identifying key evidence of eligibility.

1. Identify the limitations/constraints imposed by the technology components being utilized. What technical challenges did these constraints create?

2. Identify the degree of control the claimant has to modify the technology components. What technical challenges did these constraints create? Examples:

- Are you using any of the components in a unique, previously undocumented or unconventional fashion?

- Is the vendor able to confirm the suitability of these components for use in said fashion?

- Is the vendor capable of providing a deterministic description of the components predicted response when used in this unique fashion?

[NOTE: THE CRA FINDS THIS TYPE OF THIRD PARTY EVIDENCE VERY VALUABLE AS SUPPORTING EVIDENCE THAT THE WORK INVOLVED A "DEPARTURE FROM STANDARD PRACTICE." AS SUCH WE RECOMMEND THAT THIS EVIDENCE BE SAVED WHENEVER POSSIBLE.]

3. Identify the constraints or uncertainties or paradoxes presented when certain components/objects/technology platforms are operated in conjunction with other software entities. Do you have control over these interactions, can you or the vendors of these components predict the effects of these interactions?

4. Identify any constraints resulting from considerations of;
- Inter-operability
- Conformance to standards
- Performance (step response, throughput)
- Concurrency
- Footprint
- Scale-ability
- Stability
- 3rd party components
- legacy requirements

What technical challenges did these constraints create?

5. Identify any key characteristics of a technology platform you are using to which the manufacturer of the technology component cannot provide a fully deterministic characterization of the platform when utilized in the fashion required by your project.

6. Is the integrated performance of the software components incorporated within the project fully deterministic? I.E. can the behavior of the components be fully projected both on a stand alone basis as well as when operating within an integrated environment? Can you predict the desired outcome? If not why not?

7. What technology risks/constraints/problems appeared after the project began?

8. What was or will be hard or technically difficult to do & why?

9. What restrictions are presented by the attributes of objects/components or the API's presented by components on environmental platforms such as operating systems?

The most significant underlying key variables are:

Inter-operability methods - component selection (unresolved), Scalabity methods - memory vs. hard code (unresolved), Concurrency methods - single, dual, quad core (unresolved)

Activity #1-1: Eligible Activities (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Analysis / simulation:	2 alternatives What technical alternatives did you look at, what did you discard & why?
Process trials:	100 runs / samples Design of Experiment involves designing a set of ten to twenty experiments, in which all relevant factors are varied systematically. When the results of these experiments are analyzed, they help to identify optimal conditions, the factors that most influence the results, and those that do not, as well as details such as the existence of interactions and synergies between factors.
Physical prototypes:	3 samples (with 10 revisions) When faced with Technological Uncertainties many software developers design experiments using prototype code. Prototype code is not fully featured and is quickly built to check out critical performance issues or uncertain interfaces (Technological Uncertainties). The code is often abandoned following testing for the unknowns, but successful architectures and constructs derived through the work with the prototype are carried forward to the final design. When this design methodology is used, the claimant should have little trouble in succinctly defining the Technological Advancement attempted and the Technological Uncertainties - thus qualifying the project.

An experiment within the context of the SR&ED Program involves setting up test conditions and making observations or measurements aimed at filling gaps in our technical knowledge. The result of the experiment, whether it is successful or unsuccessful, provides an increase in knowledge of software systems relative to the Technological Advancement sought and/or the Technological Uncertainties.

The new knowledge is applicable beyond the system under test. Thus inherently, Technological Uncertainties are associated with advancements in technology knowledge. One making a claim should always be able to identify the technological advancement in his knowledge that is associated with solving a technological uncertainty, i.e. what was learned through experimentation.

In software development, immediate problems are usually solved by "trial-and-error" rather than by experiment in the sense of the Income Tax Act. Trial-and-error involves executing a series of probes that were not sequenced in a systematic pre-plan. The objective here is to resolve a functional problem (as in routine debugging) rather than to gain understandings that are expected to be more widely applicable. The lesson learned in each iteration of "trial and error" is simply "that an option didn't work" and they are not applicable in a much broader sense. For each iteration the probe is chosen that is now judged to be the most efficient in resolving the immediate problem. The process proceeds quickly from iteration to iteration.

Resolving problems through the "trial-and-error" approach is eligible support work, but it is not the basis for a Technological Advancement, as the knowledge gained does not produce a true improvement in our understanding of the technologies.

In the context of software development and the legislation, experiments might be aimed at resolving design or architectural alternatives or systematically probing an inadequately specified interface. The experimental approach itself should be designed.

Results:

- Performance (step response, throughput): 2.5 seconds (83% of goal)
- CPU load: 50 % busy (133% of goal)
- Foot Print: 5 Mb (62% of goal)

As stated in IC97-1, “experimentation or analysis in a situation where there is technological uncertainty”, is a strong indication of project eligibility. The claimant should strongly link the experimental work to the Technological Advancement sought and to the Technological Uncertainties declared in the technical description of his project. In this way, the claimant will show clearly that specific experimental work is aimed at resolving the Technological Uncertainty and thus constitutes an attempt at providing the Technological Advancement

Conclusion:

[AUTHOR'S NOTE: THE IDEAL CONCLUSIONS WOULD BRIEFLY DETAIL HOW THE RESULTS COMPARED WITH INITIAL EXPECTATIONS AND OUTLINE ANY FURTHER CONCLUSIONS WHICH COULD AFFECT FUTURE DEVELOPMENTS OF THIS NATURE.]

Related issues to illustrate via research steps & conclusions:

If you had to do it again what would you do differently?

What are the technical design trade-offs associated with these alternatives?

What are/were the possible technical outcomes other than the results you are seeking?

Documentation:

- Uploaded to RDBASE.NET: Technical documentation retained.docx (14.1KB)

Key Criteria Summary

Sample Software Industry

1100 - Software SRED - General guidelines

Benchmarks: Internet searches: 12 sites / articles
Patent searches: 43 patents
Competitive products or processes: 3 products
Potential components: 7 products
Queries to experts: 5 responses

Objectives: Performance (step response, throughput): 2 seconds
CPU load: 60 % busy
Stability : 1 # Errors/ run
Foot Print: 2 Mb

Uncertainty: 1 - Define variables for experimentation (OPTIMAL METHODS)

Key Variables: Concurrency methods - single, dual, quad core,
Inter-operability methods - component selection,
Scalabity methods - memory vs. hard code

Activity	Testing Methods	Results - % of Objective	Variables Concluded	Hours	Materials $	Subcontractor $	Fiscal Year
1 - Eligible Activities	Analysis / simulation: 2 alternatives Process trials: 100 runs / samples Physical prototypes: 3 samples ... prototype revisions: 10 revisions	Performance (step response, throughput): 2.5 seconds (83 %) CPU load: 50 % busy (133 %) Foot Print: 5 Mb (62 %)	(none)	155.50	99.00	124.00	2011

1101 - Integrate SQL Database, X-n, & Internet Servers

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
Throughput (events/s)	1	20	Yes
query rate (bit/sec)	10	1	No
data accuracy (%)	80	100	No
Storage Capacity (MB)	100	1000	No

[NOTE: THIS EXAMPLE IS REPRODUCED FROM, "GUIDANCE ON ELIGIBILITY OF SOFTWARE PROJECTS FOR THE SR&ED TAX CREDITS," AS PUBLISHED BY THE CRA IN CO-OPERATION WITH CATA & THE SOFTWARE INDUSTRY.]

[NOTE - THIS IS AN EXAMPLE OF HOW THE EVIDENCE NECESSARY TO SUPPORT THE CLAIM TYPICALLY ARISES NATURALLY FROM THE STANDARD DOCUMENTATION AND WORK PRODUCTS OF A GIVEN SOFTWARE DEVELOPMENT EFFORT.]

[AUTHOR'S NOTE: IDEALLY THE TAXPAYER WOULD ATTEMPT TO QUANTIFY THE OBJECTIVES THEY ARE TRYING TO ACHIEVE. QUANTIFIABLE OBJECTIVES HAVE BEEN ADDED ABOVE TO ILLUSTRATE.]

A large container rental company is developing a custom, geographically distributed, transaction based, enterprise wide, operations, reservations, billing, and inventory yield management system. The new system will replace an ageing and simple UNIX terminal based main-frame reservation and contract recording system.

The nature of the problem appeared in the later half of the project as a result of unexpected interactions between the transaction server component technology and the SQL database technology. Within the given architecture the two components combined to constrain the manner and mechanisms related to the level of granularity at which the SQL database could undertake record locking within a given table(s). The end result was an unexpected and severe impairment with respect to both the concurrency and throughput as it pertained to the processing of transactions.

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Patent searches	3 patents	Nothing found
Similar prior in-house technologies	1 products / processes	Existing system is a simple UNIX terminal based main-frame reservation & contract recording system.
Potential components	3 products	We looked at 4 potential components
Queries to experts	2 responses	We spoke to 2 software engineers and there is no off the counter solution

The development team contacted the vendor of the components (which in this case was common to the Operating system, SQL database, Transaction server and Internet server software technology components) and requested assistance with the problem. The vendor investigated the problem and made several recommendations in an attempt to solve the problem, but was unable to direct the company to a solution to the problem.

None of the directives from the vendor were able to correct the system performance. In fact during the course of the investigation the vendor was unable to accurately predict the resulting system performance with respect to several of the suggestions they made.

[NOTE: THIS EXPLANATION OF STANDARD PRACTICE SHOULD ATTEMPT TO OUTLINE "READILY AVAILABLE INFORMATION" ON THE TOPIC CONSIDERED AND IDENTIFY THE BOUNDARIES OF "KNOWN" AND "UNKNOWN" VARIABLES. THESE IN TURN FORM THE BASIS OF THE "TECHNICAL UNCERTAINTIES". THIS INFORMATION IS

USEFUL IN HELPING THE AUDITOR TO EVALUATE THE COMPANY'S "TECHNICAL QUALIFICATIONS" WITH RESPECT TO THE TECHNOLOGIES IN QUESTION.]

The independent consultants made the recommendation to develop rather than purchase a system as 3rd party solutions did not feature required functionality.

[EVIDENCE - PERTAINING TO THE STATE OF THE ART - IS THE REPORT EXAMINING THE SOLUTIONS AVAILABLE AND THE RECOMMENDATION TO MAKE VERSUS BUY.]

The claimant did not have the internal development expertise necessary to design and implement the new system, and consequently subcontracted a respected Canadian software development firm to undertake the project.

[EVIDENCE - RETENTION OF SUCH A DEVELOPMENT FIRM PROVIDES EVIDENCE OF ACCESS TO QUALIFIED PERSONNEL WHICH IN TURN RELATES DIRECTLY TO THE VALIDITY OF ADVANCEMENTS SOUGHT AND UNCERTAINTIES ENCOUNTERED.]

The new system architecture was implemented utilizing object oriented software technology components in an N Tier thin client configuration. The functional requirements with respect to transactional, reporting, and yield management processes for the system resulted in the requirement to support very complex transactions. This in turn required the implementation of a very large and complex database schema.

[NOTE: IDEALLY, WE WOULD TRY TO QUANTIFY THIS DATABASE ENVIRONMENT AND VARIABLES IN QUESTION].

Field of Science/Technology:

Software engineering and technology (2.02.09)

Project Details:

Intended Results:	Improve existing processes
Work locations:	Commercial Facility
Key Employees:	Bill Gates (Computers - B.Sc. (1981) / CEO), Steve Jobs (Software - B.Eng. (1983) / Project Manager), Sergery Brin (Unknown / Unknown)
Evidence types:	Test protocols, test data, analysis of test results, conclusions; Records of trial runs

Scientific or Technological Advancement:

Uncertainty #1: Database and Transaction Server Interaction

Extremely large data sets are usually quite complex, frequently containing scores of variables, many of which can only be described by non-linear relationships. Numerous variables may also interact with each other. These issues all combine to make many statistical procedures, such as Analysis of Variance or regression analysis, difficult to use. Care must also be taken such that data with many variables is not "over analyzed." Not matter how large the data set is originally, if it is cut into enough segments, significant differences will be found between groups simply by chance.

The most significant underlying key variables are:

Data handling method (conversion, number thread), Optimal data query method, Optimal database structure, Data normalization method, Data characteristic (type, size, etc)

Activity #1-1: System modeling (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Analysis / simulation:	3 alternatives The use of PLS models eliminates drawback of both Ordinary Least Squares (OLS) Regression and Principal Component Regression (PCR). OLS requires more samples (products) than variables to be included in the model, which is typically not the case when attempting to compare consumer and descriptive data. In fact, it is not unusual to present only 8-10 products to the consumer and descriptive panels yet collect information on a couple dozen descriptive attributes. PCR eliminates this

	difficulty, as well as the problem of multicollinearity that is present in much of the descriptive data. However, the first principal component, formed from the descriptive data, is not necessarily related to consumer acceptance thereby weakening the model for predictive purposes. The use of PLS through Preference Cluster Mapping eliminates both of these difficulties. Further, because it still fits a model to the data it remains possible to predict consumer liking of new products from the existing model by simply running additional descriptive panels.
Process trials:	100 runs / samples [EVIDENCE OF THE TECHNOLOGICAL UNCERTAINTY - TEST PLANS, TEST LOGS, TEST PROGRAMS, DEFECT TRACKING RECORDS, EMAIL CORRESPONDENCE WITH VENDOR(S) RELATING TO THE PROBLEM.] [EVIDENCE OF SYSTEMATIC EXPERIMENTATION - THE TEST PROTOTYPE PROGRAMS, TEST PLANS, TEST RESULTS, EMAILS, & DEFECT TRACKING ENTRIES.]
Physical prototypes:	3 samples (with 4 revisions) The development teams continued to utilize a series of prototypes and experimentation to empirically characterize the behaviour of the system in order to gain further insight into the problem. Each of the 3 potential solutions was then implemented and tested. The solution which showed the most improvement was then further refined using another series of experiments.

Results:

- Throughput: 30 events/s (152% of goal)

The development team undertook a series of testing and corrective actions but was unable to isolate the root cause for the combined interaction behaviour of the database and transaction server technology components.

Conclusion:

Discovered that relational data models could be inefficient when used in the DMS in some circumstances.

[NOTE: THE IDEAL CONCLUSION WOULD ALSO BRIEFLY DETAIL HOW THESE RESULTS COMPARED WITH INITIAL EXPECTATIONS AND OUTLINE ANY FURTHER CONCLUSIONS WHICH COULD AFFECT FUTURE DEVELOPMENTS OF THIS NATURE.]

Significant variables addressed: Data characteristic (type, size, etc), Data handling method (conversion, number thread), Data normalization method, Optimal data query method, Optimal database structure

Documentation:

- Uploaded to RDBASE.NET: software examples core_issues summary.xls (25.5KB)
- Offline Documents: Server Interaction

Key Criteria Summary

Sample Software Industry

1101 - Integrate SQL dbase, x-n, & internet servers

Benchmarks: Patent searches: 3 patents
Similar prior in-house technologies: 1 products /
Potential components: 3 products
Queries to experts: 2 responses

Objectives: Throughput: 20 events/s
query rate: 1 bit/sec
data accuracy : 100 %
Storage Capacity : 1000 MB

Uncertainty: 1 - Database and Transaction Server Interaction

Key Variables: Data characteristic (type, size, etc), Data handling method (conversion, number thread), Data normalization method, Optimal data query method, Optimal database structure

Activity	Testing Methods	Results - % of Objective	Variables Concluded	Hours	Materials $	Subcontractor $	Fiscal Year
1 - System modeling	Analysis / simulation: 3 alternatives Process trials: 100 runs / samples Physical prototypes: 3 samples ... prototype revisions: 4 revisions	Throughput: 30 events/s (152 %)	Data characteristic (type, size, etc) Data handling method (conversion, number thread) Data normalization method Optimal data query method Optimal database structure	425.00	4,263.00	3.00	2011

1102 – DBMS Project (Database Methodology)

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
Increase access speed (s)	30	15	Yes
Increase database capacity (GB)	0.5	1	No

[NOTE: THIS PROJECT DESCRIPTION IS BASED ON THE CRA'S EXAMPLE OF AN ELIGIBLE PROJECT FROM THEIR SR&ED SOFTWARE DEVELOPMENT INDUSTRY GUIDELINES: INFORMATION CIRCULAR 97-1.]

The objective is to develop and implement a new data basing method in order to double the speed of the database currently achieved in Version 3.5 of our "property record management system."

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Internet searches	21 sites / articles	No results related to our query
Patent searches	14 patents	Searched Google patents
Similar prior in-house technologies	1 products / processes	Existing system has excessive access times (>30 seconds) with large databases (>1 gigabyte).

XYZ Co. has developed a proprietary DMS (database management system) as part of their PRMS (property record management system) product. The DMS works well with small data sets, but has excessive access times (>30 seconds) with large databases (>1 gigabyte).

[NOTE: THIS EXPLANATION OF STANDARD PRACTICE SHOULD ATTEMPT TO OUTLINE "READILY AVAILABLE INFORMATION" ON THE TOPIC CONSIDERED AND IDENTIFY THE BOUNDARIES OF "KNOWN" AND "UNKNOWN" VARIABLES. THESE IN TURN FORM THE BASIS OF THE "TECHNICAL UNCERTAINTIES". THIS INFORMATION IS USEFUL IN HELPING THE AUDITOR TO EVALUATE THE COMPANY'S "TECHNICAL QUALIFICATIONS" WITH RESPECT TO THE TECHNOLOGIES IN QUESTION.]

Field of Science/Technology:

Software engineering and technology (2.02.09)

Project Details:

Intended Results:	Develop new materials, devices, or products, Improve existing materials, devices, or products
Work locations:	Commercial Facility
Key Employees:	Bill Gates (Computers - B.Sc. (1981) / CEO), Steve Jobs (Software - B.Eng. (1983) / Project Manager), Sergery Brin (Unknown / Unknown)
Evidence types:	Test protocols, test data, analysis of test results, conclusions; Design of experiments; Records of trial runs

Scientific or Technological Advancement:

Uncertainty #1: Optimal data model

Extremely large data sets are usually quite complex, frequently containing scores of variables, many of which can only be described by non-linear relationships. Numerous variables may also interact with each other. These issues all combine to make many statistical procedures, such as Analysis of Variance or regression analysis, difficult to use. Care must also be taken such that data with many variables is not "over analyzed." No matter how large the data set is originally, if it is cut into enough segments, significant differences will be found between groups simply by chance.

[AUTHORS NOTE]: Relational Data Model Analysis - We were unsure regarding the optimal synchronization of the various databases and the negative result that a rational data model, and various datatypes might have with DMS.

Comm model vs relational environment - We were uncertain how we will use a data model designed for data communications in a relational environment?

Relational Access + Packet Access Combination - How can we optimally combine relational and packet access against the same database to yield a minimum # of inefficiencies when processing tables in the DMS?

The most significant underlying key variables are:

Data model selection (relational, comm), Optimal query method (relational, packet), Data size/type, Data normalization method

Activity #1-1: Data Communications Model Analysis (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Analysis / simulation:	4 alternatives Conducted a literature review of relational data models. As a result we looked at 4 alternate data models. Conducted 7 comprehensive benchmark tests to compare performance between the the models. [NOTE: IDEALLY, CLAIMANTS WOULD PROVIDE SPECIFIC DETAILS AS TO HOW THESE MODELS DIFFERED AND SOME OF THE MOST SIGNIFICANT VARIABLES EXAMINED. IN ADDITION TO A BRIEF OVERVIEW OF THE WORK PERFORMED EACH ACTIVITY SHOULD ATTEMPT TO CROSS-REFERENCE RELEVANT, TECHNICAL DOCUMENTATION INCLUDING: DOCUMENT NAME, DATE, # OF PAGES AND LOCATION.]
Process trials:	10 runs / samples We experimented to determine if an existing data communications model could be adapted to achieve processing efficiencies, at the expense of additional storage space. [NOTE: IDEALLY CLAIMANTS WOULD PROVIDE SPECIFIC DETAILS AS TO HOW THESE DATABASES DIFFERED AND WHY THIS WAS BELIEVED TO BE TECHNICALLY SIGNIFICANT. WE SHOULD ALSO ATTEMPT TO SUMMARIZE SOME OF THE MOST SIGNIFICANT VARIABLES EXAMINED.]
Physical prototypes:	1 samples Experimentally employed a hybrid approach involving both relational and packet data management techniques. Created a prototype Data Model DMS with the intention of making it faster than existing one.

Results:

- Increase access speed : 10 s (133% of goal) -- Processing time for query and update capabilities were improved to <10 seconds for >1 GB databases.

Initial testing indicated that the new DMS was 75% faster than the existing DMS through use of the newly developed hybrid data access techniques.

Conclusion:

We determined that a data communications model could achieve processing inefficiencies. We discovered that relational data models could be inefficient when used in the DMS in some circumstances. While some of the tables could be processed more efficiently if they were in packet form, others were best managed through relational techniques.

[NOTE: THE IDEAL CONCLUSION WOULD ALSO BRIEFLY DETAIL HOW THESE RESULTS COMPARED WITH INITIAL EXPECTATIONS AND OUTLINE ANY FURTHER CONCLUSIONS WHICH COULD AFFECT FUTURE DEVELOPMENTS OF THIS NATURE.]
[NOTE: IDEALLY WE WOULD OUTLINE ADDITIONAL DETAILS SUCH AS "PROS AND CONS" DISCOVERED WITH RESPECT TO THIS METHOD - PARTICULARLY THOSE THAT WERE OTHERWISE UNEXPECTED.]

Significant variables addressed: Data model selection (relational, comm), Data normalization method, Data size/type, Optimal query method (relational, packet)

Documentation:

- Offline Documents: Test results

Key Criteria Summary

Sample Software Industry

1102 - DBMS Project (Database methodology)

Benchmarks: Internet searches: 21 sites / articles
Patent searches: 14 patents
Similar prior in-house technologies: 1 products /

Objectives: Increase access speed : 15 s
Increase database capacity : 1 GB

Uncertainty: 1 - Optimal data model

Key Variables: Data model selection (relational, comm), Data normalization method, Data size/type, Optimal query method (relational, packet)

Activity	Testing Methods	Results - % of Objective	Variables Concluded	Hours	Materials $	Subcontractor $	Fiscal Year
1 - Data Communications Model Analysis	Analysis / simulation: 4 alternatives Process trials: 10 runs / samples Physical prototypes: 1 samples	Increase access speed : 10 s (133 %)	Data model selection (relational, comm) Data normalization method Data size/type Optimal query method (relational, packet)	600.00	0.00	2,500.00	2011

1103 - Network Failure Problems

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
Concurrent accesses (# transactions)	760	1750	Yes
Reduce response time (seconds)	5	3	Yes
Network performance (# crashes/10 transactions)	1	0	Yes
Development costs ($)	500	400	Yes

[THIS PROJECT IS BASED ON THE CRA'S EXAMPLE #3 FROM, "CROSS-SECTOR SHOP FLOOR GUIDANCE DOCUMENT" (JULY 29, 2002)]

[AUTHOR'S NOTE: IDEALLY THE TAXPAYER WOULD ATTEMPT TO QUANTIFY THE OBJECTIVES THEY ARE TRYING TO ACHIEVE. QUANTIFIABLE OBJECTIVES HAVE BEEN ADDED ABOVE TO ILLUSTRATE.]

The objective was to determine why the CallHome high-speed network does not meet the original design criteria, and to take correction such that the network will facilitate 500 high-speed access ports, with 1750 concurrent transactions, at a maximum 25% reduction in response time.

CallHome's technological advancement sought was the solution to the network failure. The network had been designed according to the current theory, and had failed to provide the theoretical performance. Once normal network troubleshooting proved ineffective, and did not solve the problems, CallHome realized more design work was required to determine the underlying problem with the technology. Solving this problem represented a technological advancement.

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Internet searches	12 sites / articles	Nothing matched our query
Patent searches	4 patents	Searched Google patents, we looked at 4 simlar products that didn't meet our specs
Similar prior in-house technologies	1 products / processes	Theoretical capacity was 1750 concurrent transactions, but system was crashing in practice.

A communications company, CallHome, designed and built a high-speed Internet access network to offer its clients. Three different vendors' equipment were involved in the implementation of the network, including the local telephone company equipment. The initial design was capable of 500 high-speed ports. Maximum theoretical network capacity was established as 1750 concurrent transactions, with a maximum 25% reduction in response time. After reviewing the overall network design, all the equipment vendors agreed that their equipment could operate in the target network architecture, and these numbers could be comfortably achieved.

After six months of operation, CallHome had sold 225 high-speed accesses, and the network management system was reporting utilization numbers of 700-800 concurrent accesses, with an 18% reduction in response rate. Although this reduction in response time raised concerns about network capacity, there were no customer complaints, and the vendors continued to stand by their position. CallHome then had a major marketing campaign that resulted in another 60 high-speed customers.

After their service was activated, the network began crashing for no apparent reason. The network management software could not pinpoint the problem, and the equipment vendors could offer no reasons for the failures. Customers began canceling their service.

Field of Science/Technology:

Computer hardware and architecture (2.02.08)

Project Details:

Intended Results:	Improve existing materials, devices, or products
Work locations:	Commercial Facility
Key Employees:	Bill Gates (Computers - B.Sc. (1981) / CEO), Steve Jobs (Software - B.Eng. (1983) / Project Manager), Sergery Brin (Unknown / Unknown)
Evidence types:	Samples, prototypes, scrap or other artefacts; Design, system architecture and source code; Project records, laboratory notebooks; Project planning documents

Scientific or Technological Advancement:

Uncertainty #1: System Uncertainty

[NOTE: WHICH VARIABLES ARE UNPREDICTABLE WITH RESPECT TO DETERMINING THE "OPTIMAL COMBINATION OF COMPONENTS"? THESE ISSUES ARE CREATED BY DEPARTURES FROM STANDARD PRACTICES.]

-What method should be used to determine the cause of the network failure, given the network management software, the vendors, and all other network indicators are normal?
-The vendors would not provide detailed information on their firmware source code. Their equipment complies with network and protocol interface standards. Can we, and how do we, develop a test bed that will provide critical technical information necessary to identify the technical problems?
-Once the inconsistencies in networking address index caching were identified, it was technically uncertain how we could develop an interface that will compensate for the different index caching techniques, without compromising on response times.

The extent of system uncertainty of possible resolutions to problems is unknown. For instance, a possible solution may solve one component but may also cause the other components to fail. Therefore, it is possible that a solution cannot be developed to address the needs of all the components. For example, a software solution of Vendor A may not correctly interface with a specific piece of firmware from Vendor B and C.

The most significant underlying key variables are:

Optimal system protocol, Inter-operability/integration of components, Network architectural design vs various equipment, Index caching methods, Data normalization methods (unresolved)

Activity #1-1: Development and Testing (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Analysis / simulation:	4 alternatives
	4 architectural designs experimentally evaluated. All designs failed, which resulted in a completely new approach that involved developing new techniques for disparate data normalization.
Process trials:	32 runs / samples
	[AUTHOR'S NOTE: THE DESCRIPTIONS BELOW WERE PROVIDED IN THE CRA'S EXAMPLE. THE DATA ABOVE (# TRIALS/ALTERNATIVES) IS PROVIDED TO ILLUSTRATE SOME OF THE ADDITIONAL DETAILS THAT WOULD IDEALLY BE INCLUDED.]
	[NOTE: TRY TO CLARIFY VARIABLES IN QUESTION AND ILLUSTRATE ANY UNEXPECTED INTER-RELATIONS.]
	Engineers and other technical staff studied and analyzed the problem. Our preliminary conclusions led us to speculate that the problem happens when the user community demands 760 concurrent transactions.
	Conducted a number [HOW MANY?] of different experiments on the network system management raw data. From the analysis we determined that the network management system was not properly reporting on the interface conditions between vendor A and vendor B equipment.
	Developed and built a number [HOW MANY?] of different network management tools to look at how the equipment from vendor A and B were interacting with each other. Analysis of implementing the tool revealed that the two vendors were using

	slightly different network address caching methods. Three different units from each vendor were tested, all with the same results of a crashed network in our test bed simulations. Both vendors refused us access to their firmware source code, and reported there was no problem with their equipment. They stated that their equipment conformed to OSI network and protocol interface standards, so that access to the source code was not necessary. We began to research different techniques for integrating the different vendor equipment without the performance reduction we were experiencing in our current network.
Physical prototypes:	1 samples (with 10 revisions) We built a prototype interface unit and evaluated it in our test bed. We developed 7 tools to look at equipment interaction, 3 units from each of 2 vendors tested. We made 10 modifications on network management system.

Results:

- Concurrent accesses: 1900 # transactions (115% of goal)
- Reduce response time: 4.1 seconds (45% of goal)
- Network performance : 0 # crashes/10 transactions (100% of goal)
- Development costs: 450 $ (50% of goal)

CallHome developed a new set of integration firmware that not only resolved the original firmware disparities, it introduced new techniques for networking index caching that theoretically pushed the CallHome network capability to 1900 concurrent access requests, with an 18% reduction in response time. Also, the company attempted to develop system integration techniques, which unfortunately failed because they did not improve reliability.

Conclusion:

CallHome learned that the different vendor's equipment used different techniques for indexing and routing of network addresses. A simpler model makes the identification and creation of an "ideal" product much more direct as fewer variables need to be examined for their effect on consumer acceptance. Development costs are also lowered as descriptive panels only need to collect information on a handful of attributes for use in predicting consumer acceptance of these new products.

[NOTE: IDEALLY, WE NEED COMPARISONS TO INITIAL EXPECTATIONS & TO CONCLUDE ON THE UNCERTAINTIES STATED: I.E. WHAT IF ANY ARCHITECTURES WOULD ACCOMPLISH THE COMPRESSION OBJECTIVES AND WHY? IT IS THE HYPOTHESES OR CONCLUSIONS TO EXPLAIN THESE RESULTS, RATHER THAN THE RESULTS THEMSELVES, WHICH THE CCRA WISHES TO SEE EVIDENCE OF.]

Significant variables addressed: Index caching methods, Inter-operability/integration of components, Network architectural design vs various equipment, Optimal system protocol

Documentation:

- Offline Documents: Network Setup

Key Criteria Summary

Sample Software Industry

1103 - Network failure problems

Benchmarks: Internet searches: 12 sites / articles
Patent searches: 4 patents
Similar prior in-house technologies: 1 products /

Objectives: Concurrent accesses: 1750 # transactions
Reduce response time: 3 seconds
Network performance : 0 # crashes/10 transactions
Development costs: 400 $

Uncertainty: 1 - System Uncertainty

Key Variables: Data normalization methods, Index caching methods, Inter-operability/integration of components, Network architectural design vs various equipment, Optimal system protocol

Activity	Testing Methods	Results - % of Objective	Variables Concluded	Hours	Materials $	Subcontractor $	Fiscal Year
1 - Development and Testing	Analysis / simulation: 4 alternatives Process trials: 32 runs / samples Physical prototypes: 1 samples ... prototype revisions: 10 revisions	Concurrent accesses: 1900 # transactions (115 %) Development costs: 450 $ (50 %) Reduce response time: 4.1 seconds (45 %) Network performance : 0 # crashes/10 transactions (100 %)	Index caching methods Inter-operability/integration of components Network architectural design vs various equipment Optimal system protocol	159.25	8.00	66.00	2011

Project Name:	Compression algorithm	**Start Date:**	2009-01-01
Project Number:	1104	**Completion Date:**	2015-04-30

1104 - Compression Algorithm

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
Compressed size of 1 MB map (KB)	90	30	Yes
Minimize data loss (%)	3	1.9	Yes
Minimize CPU load (% busy)	35	30	Yes
Maintain quality (peak signal-to-noise ratio) (dB)	30	30	Yes

[THIS EXAMPLE IS REPRODUCED FROM "GUIDANCE ON ELIGIBILITY OF SOFTWARE PROJECTS FOR THE SR&ED TAX CREDITS," AS PUBLISHED BY THE CRA IN CO-OPERATION WITH CATA & THE SOFTWARE INDUSTRY.]

The objective was to develop a new compression tool for GIS information with the capability of compressing a 1 meg map down to 30K. This has to be accomplished with less than 2% data loss.

Our product is very similar to that of ABC-IT Inc, however due to limited memory and battery life only limited size maps could be loaded, and a limited number of notes could be attached. Our competitor has released their software with a 50% further decrease in their compressed maps, i.e. they can compress a 1 meg. map to less than 40K. Our current best compression is to get a 1 meg image down to 90K. In order to maintain market share we must at least meet their performance, and develop a new compression technique. Our eventual goal is to be down to 30K.

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Internet searches	5 sites / articles	Found 5 website that had 13 articles, nothing matched our criteria
Patent searches	3 patents	Searched Google patents
Competitive products or processes	1 products	Competitor can compress 1Mb map to <40K.
Similar prior in-house technologies	1 products / processes	Our current product can compress 1Mb to 90K.
Potential components	4 products	We looked at 4 potential components
Queries to experts	2 responses	We spoke to 2 software engineers and there is no readily available solution

In September of last year our competitor ABC-IT Inc. released a new tool suite for the compression and modification of electronic maps and overlays. The tools are designed for use on mobile platforms. They allow the user to make notes, and modify the electronic maps as they are doing field work. They can then upload the changes to their desktop PC, back at the office, where a full GIS package resides.

Field of Science/Technology:

Software engineering and technology (2.02.09)

Project Details:

Intended Results:	Develop new materials, devices, or products
Work locations:	Commercial Facility
Key Employees:	Bill Gates (Computers - B.Sc. (1981) / CEO), Steve Jobs (Software - B.Eng. (1983) / Project Manager), Sergery Brin (Unknown / Unknown)
Evidence types:	Test protocols, test data, analysis of test results, conclusions; Project records, laboratory notebooks; Design of experiments

Project Name:	Compression algorithm	**Start Date:**	2009-01-01
Project Number:	1104	**Completion Date:**	2015-04-30

Scientific or Technological Advancement:

Uncertainty #1: Optimal compression method

Optimal compression method(s). Specifically, what, if any, architectures would accomplish this compression objective?

The most significant underlying key variables are:

Compression architecture, Scalability method, Image characteristics (maps, overlays)

Activity #1-2: Development and Testing (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Physical prototypes:	4 samples

[AUTHOR'S NOTE: THE DESCRIPTIONS BELOW WERE PROVIDED IN THE CRA'S EXAMPLE. THE DATA ABOVE (# TRIALS/ALTERNATIVES) IS PROVIDED TO ILLUSTRATE SOME OF THE ADDITIONAL DETAILS THAT WOULD IDEALLY BE INCLUDED.]

Through development and experimentation with several approaches [AUTHOR'S NOTE: IDEALLY THE DESCRIPTION WOULD DETAIL HOW MANY PROTOTYPE VARIATIONS DID WE ATTEMPT? I.E. 5, 50, 500? WERE THEY ALL SIMILAR OR COMPLETELY DIFFERENT? IF DIFFERENT, HOW SO AND WHY?], we managed to develop a compression tool using a data communication standard (X2 standard for hardware compression), and a method of analyzing the maps and overlays, synchronizing them into a single image and then using a modified version of MPEG 3 compression.

The modified software compression allows for easier separation of the map from the overlay once the data is transferred from the hand held unit to the desktop PC.

[NOTE: THIS DESCRIPTION IS STILL FAIRLY WEAK IN THAT THE "ACTIVITIES" & "CONCLUSIONS" ARE CURRENTLY BASED TOO HEAVILY ON A "GOALS - RESULTS" ORIENTATION RATHER THAN ILLUSTRATING WHY IT WAS SO HARD TO GET TO THE FINAL SOLUTION AND THEN IDENTIFYING RELEVANT "TECHNICAL CONCLUSIONS."]

Results:

- Compressed size of 1 MB map: 30 KB (100% of goal)
- Minimize data loss: 1.8 % (109% of goal)
- Minimize CPU load: 32 % busy (60% of goal)
- Maintain quality (peak signal-to-noise ratio): 30 dB (100% of goal)

The main issues were obtaining sufficient compression and allowing separation of the map and overlays.

Conclusion:

ELIGIBLE AS WRITTEN:

According to the CRA, "Generally, this Advancement WOULD QUALIFY, BUT it would NOT qualify in either of the following two situations."

[INELIGIBLE IF:]

1. While doing the preliminary technical feasibility work we discovered a company in the US that has a tool ideally suited for our needs. We are currently working out a licensing agreement for resale. With a couple of parameter changes, their tool will give us our target compression.

[RATIONALE: A ROUTINE SOLUTION WAS FOUND AND IMPLEMENTED WITHOUT SYSTEM UNCERTAINTY WITH RESPECT TO THE OPTIMAL METHOD(S) OF INTEGRATION.]

OR,

2. In the early part of the technical feasibility study portion of the project, we learned that one of the senior software engineers had resigned from ABC-IT Inc. We hired him and he is redeveloping their algorithm for our application. We have decided that matching the ABC-IT Inc. performance will be adequate.

[RATIONALE: THOUGH THE DEVELOPMENT MAY HAVE BEEN ELIGIBLE FOR THE COMPANY, HIRING THE NEW EMPLOYEE IMMEDIATELY EXPANDED ITS "STANDARD PRACTICE KNOWLEDGE BASE" TO INCLUDE THIS EMPLOYEE'S KNOWLEDGE. SINCE THE SOLUTION COULD BE DEVELOPED BY HIM WITHOUT ANY FURTHER

"TECHNOLOGICAL UNCERTAINTY" IT DOES NOT REPRESENT AN ELIGIBLE EXPERIMENTAL DEVELOPMENT ACTIVITY.]

[NOTE: IDEALLY, WE NEED COMPARISONS TO INITIAL EXPECTATIONS & TO CONCLUDE ON THE UNCERTAINTIES STATED: I.E. WHAT IF ANY ARCHITECTURES WOULD ACCOMPLISH THE COMPRESSION OBJECTIVES AND WHY? IT IS THE HYPOTHESES OR CONCLUSIONS TO EXPLAIN THESE RESULTS, RATHER THAN THE RESULTS THEMSELVES, WHICH THE CRA WISHES TO SEE EVIDENCE OF.]

Significant variables addressed: Compression architecture, Image characteristics (maps, overlays), Scalability method

Documentation:

- Offline Documents: Test results

Key Criteria Summary

Sample Software Industry

1104 - Compression algorithm

Benchmarks:
Internet searches: 5 sites / articles
Patent searches: 3 patents
Competitive products or processes: 1 products
Similar prior in-house technologies: 1 products /
Potential components: 4 products
Queries to experts: 2 responses

Objectives:
Compressed size of 1 MB map: 30 KB
Minimize data loss: 1.9 %
Minimize CPU load: 30 % busy
Maintain quality (peak signal-to-noise ratio): 30 dB

Uncertainty: 1 - Optimal compression method

Key Variables: Compression architecture, Image characteristics (maps, overlays), Scalability method

Activity	Testing Methods	Results - % of Objective	Variables Concluded	Hours	Materials $	Subcontractor $	Fiscal Year
2 - Development and Testing	Physical prototypes: 4 samples	Compressed size of 1 MB map: 30 KB (100 %) Minimize data loss: 1.8 % (109 %) Minimize CPU load: 32 % busy (60 %) Maintain quality (peak signal-to-noise ratio): 30 dB (100 %)	Compression architecture Image characteristics (maps, overlays) Scalability method	81.25	2,000.00	5,000.00	2011

1105 – Scaling vs. Speed vs. Compression

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
Encoding rate (s/GB)	20	5	Yes
File compression ratio (:1)	2	3	Yes
Maximum size (GB)	100	200	Yes

[THIS EXAMPLE IS REPRODUCED FROM "GUIDANCE ON ELIGIBILITY OF SOFTWARE PROJECTS FOR THE SR&ED TAX CREDITS," AS PUBLISHED BY THE CRA IN CO-OPERATION WITH CATA & THE SOFTWARE INDUSTRY.]

[AUTHOR'S NOTE: IDEALLY THE TAXPAYER WOULD ATTEMPT TO QUANTIFY THE OBJECTIVES THEY ARE TRYING TO ACHIEVE. A QUANTIFIABLE OBJECTIVE HAS BEEN ADDED ABOVE, TO ILLUSTRATE.]

We seek to show through analysis that the key to both graceful scaling to higher speed platforms and speed maximization for a specified compression on a Pentium performing "Framis" coding is the optimal use of the Pentium look ahead.

We also want to analytically determine the compression expected as a function of the number of bytes processed in parallel and to thereby determine the speed/compression trade-off.

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Competitive products or processes	1 products	Competitor capable of encoding a gigabyte in seven seconds on a 400 Mhz Pentium III processor.
Potential components	7 products	There are seven potential components we are looking at using

In September of last year our competitor, MedsInc, announced its Framis encoding software capable of encoding a gigabyte in seven seconds on a 400 Mhz Pentium III processor.

[NOTE: AN IDEAL DESCRIPTION WOULD ALSO OUTLINE WHAT METHODS WERE "PUBLIC" VS. "PROPRIETARY INFORMATION" SINCE THE RESEARCHER IS EXPECTED TO TAKE NOTE OF "READILY AVAILABLE INFORMATION" BUT IS NOT EXPECTED TO KNOW "INFORMATION PROPRIETARY TO A COMPETITOR OR KNOWN IN ONLY SPECIALIST OR ACADEMIC CIRCLES."]

Field of Science/Technology:

Software engineering and technology (2.02.09)

Project Details:

Intended Results:	Improve existing processes, Improve existing materials, devices, or products
Work locations:	Commercial Facility
Key Employees:	Bill Gates (Computers - B.Sc. (1981) / CEO), Steve Jobs (Software - B.Eng. (1983) / Project Manager), Sergery Brin (Unknown / Unknown)
Evidence types:	Samples, prototypes, scrap or other artefacts; Design, system architecture and source code; Project planning documents

Scientific or Technological Advancement:

Uncertainty #1: Determining optimal trade-off

The challenge is analyzing and determining the optimal trade-off between scaling, speed and compression on several computing platforms.

[NOTE: IDEALLY WE WOULD CLARIFY WHICH VARIABLES ARE UNPREDICTABLE WITH RESPECT TO DETERMINING THE "OPTIMAL COMBINATION OF COMPONENTS"? THESE ISSUES ARE CREATED BY DEPARTURES FROM STANDARD PRACTICES.]

The most significant underlying key variables are:

Scaling method, Compression method, Computing platform

Activity #1-2: Encoding algorithm (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Process trials:	52 runs / samples 8 tests on each of 4 platforms to determine trade-off between scaling, speed & compression, followed by testing of new algorithm by 5 samples on each of the 4 platforms.

[AUTHOR'S NOTE: THE DESCRIPTIONS BELOW WERE PROVIDED IN THE CRA'S EXAMPLE. THE DATA ABOVE (# TRIALS/ALTERNATIVES) IS PROVIDED TO ILLUSTRATE SOME OF THE ADDITIONAL DETAILS THAT WOULD IDEALLY BE INCLUDED.]

During the project the company used a combination of routine methods to analyze the trade-off between scaling, speed and compression on several computing platforms.
[NOTE: THE OPTIMAL DESCRIPTION WOULD DETAIL HOW MANY PROTOTYPE VARIATIONS ATTEMPTED (5?, 50?, 500?) AND SIGNIFICANCE OF ANY DIFFERENCES]

Next we performed testing to determine whether or not the new encoding algorithm consistently met its target of being able to code a gigabyte in five seconds. This involved the testing of performance against several gigabyte samples and the writing of a report that described performance as a function of the properties of the specific data samples. [HOW MANY TESTS PERFORMED?]

[NOTE: TRY TO CLARIFY VARIABLES IN QUESTION AND ILLUSTRATE ANY UNEXPECTED INTER-RELATIONS.]

Results:

- Encoding rate: 5 s/GB (100% of goal)
- File compression ratio: 2.5 :1 (50% of goal)
- Maximum size: 110 GB (10% of goal)

We were able to achieve our speed goals but not compression or scalability objectives

Conclusion:

[NOTE - THE IDEAL SOLUTION WOULD COMPARE RESULTS TO INITIAL EXPECTATIONS AND TRY TO PROVIDE ADDITIONAL TECHNICAL EXPLANATIONS WITH RESPECT TO THE STATE UNCERTAINTIES: OPTIMAL TRADE-OFF BETWEEN SCALING, SPEED AND COMPRESSION ON SEVERAL COMPUTING PLATFORMS.]

ELIGIBLE ACTIVITIES CUT-OFF:
In the CRA's view the company showed that the new Framis had to be interfaced to an existing data input system and that a new graphical user interface had to be developed to determine ?whether or not the new Framis would scale to higher speed platforms more gracefully" and therefore the costs of the supporting activities, although routine in themselves, are eligible.

The project ends when the Framis software is sufficiently debugged and sufficiently featured that it codes a gigabyte in less than five seconds on a 400 MHz Pentium III and it can be shown that it scales to at least a couple platforms with clocks in excess of 400 MHz in a manner which is more "graceful" than was possible with the competitor's algorithm.

[NOTE: THIS POINT CAN BE EXTENDED IF THE COMPANY CAN EXPAND THE SCOPE OF THE TECHNICAL OBJECTIVE].

Significant variables addressed: Compression method, Computing platform, Scaling method

Documentation:

- Offline Documents: Algorithm

Key Criteria Summary

Sample Software Industry

1105 - Scaling vs. speed vs. compression

Benchmarks: Competitive products or processes: 1 products
Potential components: 7 products

Objectives: Encoding rate: 5 s/GB
File compression ratio: 3 :1
Maximum size: 200 GB

Uncertainty: 1 - Determining optimal trade-off

Key Variables: Compression method, Computing platform, Scaling method

Activity	Testing Methods	Results - % of Objective	Variables Concluded	Hours	Materials $	Subcontractor $	Fiscal Year
2 - Encoding algorithm	Process trials: 52 runs / samples	File compression ratio: 2.5 :1 (50 %) Maximum size: 110 GB (10 %) Encoding rate: 5 s/GB (100 %)	Scaling method Compression method Computing platform	150.00	500.00	2,500.00	2011

1106 – Software Data Warehouse Development

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
CPU Utilization (% busy)	95	70	Yes
Response Time (seconds)	60	15	Yes
Data to compression (:1 ratio)	5	15	Yes

[NOTE: THIS EXAMPLE IS REPRODUCED FROM, "GUIDANCE ON ELIGIBILITY OF SOFTWARE PROJECTS FOR THE SR&ED TAX CREDITS," AS PUBLISHED BY THE CRA IN CO-OPERATION WITH CATA & THE SOFTWARE INDUSTRY.]

[AUTHOR'S NOTE: IDEALLY THE TAXPAYER WOULD ATTEMPT TO QUANTIFY THE OBJECTIVES THEY ARE TRYING TO ACHIEVE. QUANTIFIABLE OBJECTIVES HAVE BEEN ADDED ABOVE TO ILLUSTRATE.]

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Internet searches	33 sites / articles	33 sites & 14 resulting artciles reviewed
Patent searches	4 patents	Searched Google patents, we looked at 4 simlar products that didn't meet our specs
Competitive products or processes	6 products	no methods to characterize non-uniform, dynamic data
Similar prior in-house technologies	2 products / processes	Benchmarks for CPU utilization
Potential components	100 products	over 100 potential components reviewed (open source & proprietary)
Queries to experts	3 responses	Queries to experts: 3 responses -- no methods to characterize non-uniform, dynamic data

Field of Science/Technology:

Computer hardware and architecture (2.02.08)

Project Details:

Intended Results:	Improve existing processes
Work locations:	Commercial Facility
Key Employees:	Bill Gates (Computers - B.Sc. (1981) / CEO), Steve Jobs (Software - B.Eng. (1983) / Project Manager), Sergery Brin (Unknown / Unknown)
Evidence types:	Records of resources allocated to the project, time sheets; Samples, prototypes, scrap or other artefacts; Design of experiments; Project planning documents; Project records, laboratory notebooks; Records of trial runs; Contracts

Scientific or Technological Advancement:

Uncertainty #1: Non-uniform dataset determination

We are uncertain as to how and whether it is possible to develop a method to identify and exploit the unique properties of non-uniform data sets. We are also uncertain whether we can use compressed data blocks vs entire tables to traverse the database and how much of a performance improvement this will result in.

The most significant underlying key variables are:

Definition and construction of data blocks, Methods to characterize non-uniform data, Definition of construction of data blocks, Data compression method, Data normalization method

Activity #1-1: Develop generic data model (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Analysis / simulation:	10 alternatives
	Identified most common frequency values & evaluated use of column value frequencies to create prototype compression dictionary - using a relational dbase environment

Results:

No results have been recorded for this Activity.

At the end of this first phase we found that a reasonably accurate data set model could be created. This was further tested and the data set model accuracy was verified and validated against several concrete smaller-sized relational databases available to us in the data warehouse.

Conclusion:

Model proved feasible - developed table-wide list of most frequent values for compression dictionary

Significant variables addressed: Definition and construction of data blocks

Documentation:

- Offline Documents: CRA Software Guidelines

Activity #1-2: Develop compression methods (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Physical prototypes:	10 samples
	Developed test scripts to compared CPU utilization, integrity and data throughput for operations including: parallel load, delete/update operations, full table scan & access by row.

Results:

No results have been recorded for this Activity.

Conclusion:

We determined it is best to restrict query/refresh options to compressed blocks vs. entire tables

Significant variables addressed: Definition of construction of data blocks

Documentation:

- Offline Documents: Compression algorithm

Activity #1-3: Compression algorithm with dynamic techniques (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Physical prototypes:	3 samples (with 12 revisions)
	Examined use of buffer cache to organize & control compression dictionaries when calls made to uncompress multiple blocks

Results:

- CPU Utilization: 66 % busy (116% of goal)
- Response Time: 22 seconds (84% of goal)
- Data to compression: 131 :1 ratio (1260% of goal)

In August 2008, a final prototype was selected for widespread commercial implementation ending this aspect of the experimental development.

Conclusion:

This development lead to the discovery that we could use the column value frequency of initial tables rows to create an effective block-based compression dictionary.

Significant variables addressed: Data normalization method, Definition and construction of data blocks, Definition of construction of data blocks

Documentation:

- Offline Documents: x

Activity #1-4: Extend data compression methods (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Process trials:	102 runs / samples
	Used external consultant - exploration into use of the implemented compression prototype for data backup and recovery operations

Results:

No results have been recorded for this Activity.

As the result of this work it was found out and further documented that the prototype provided measurable performance improvements [QUANTIFY] when applied to very large databases in excess of 2.5 million rows (1.3 GB) such as those typically encountered in data warehouses.

Conclusion:

Success attributed primarily to compression dictionary vs. data blocks

Significant variables addressed: Data compression method, Definition and construction of data blocks

Documentation:

- Offline Documents: Results

Activity #1-5: Correlate compression block size with initial data set (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Analysis / simulation:	22 alternatives
	the implemented prototype was used to determine whether or not an optimal data table compression-block size could be determined by both the initial data set analysis and the dynamic

Results:

No results have been recorded for this Activity.

Conclusion:

Could NOT correlate compression block size w initial data set & dynamic analysis

Significant variables addressed: Definition and construction of data blocks, Definition of construction of data blocks, Methods to characterize non-uniform data

Documentation:

- Offline Documents: Contract

Key Criteria Summary

Sample Software Industry

1106 - Software Data Warehouse Development

Benchmarks: Internet searches: 33 sites / articles
Patent searches: 4 patents
Competitive products or processes: 6 products
Similar prior in-house technologies: 2 products /
Potential components: 100 products
Queries to experts: 3 responses

Objectives: CPU Utilization: 70 % busy
Response Time: 15 seconds
Data to compression: 15 :1 ratio

Uncertainty: 1 - Non-uniform dataset determination

Key Variables: Data compression method, Data normalization method, Definition and construction of data blocks, Definition of construction of data blocks, Methods to characterize non-uniform data

Activity	Testing Methods	Results - % of Objective	Variables Concluded	Hours	Materials $	Subcontractor $	Fiscal Year
1 - Develop generic data model	Analysis / simulation: 10 alternatives	(none)	Definition and construction of data blocks	735.00	850.00	13,966.00	2011
2 - Develop compression methods	Physical prototypes: 10 samples	(none)	Definition of construction of data blocks	700.00	600.00	0.00	2011
3 - Compression algorithm with dynamic techniques	Physical prototypes: 3 samples ... prototype revisions: 12 revisions	Response Time: 22 seconds (84 %) Data to compression: 131 :1 ratio (1260 %) CPU Utilization: 66 % busy (116 %)	Definition and construction of data blocks Data normalization method Definition of construction of data blocks	695.00	6,000.00	2,550.00	2011
4 - Extend data compression methods	Process trials: 102 runs / samples	(none)	Data compression method Definition and construction of data blocks	1,230.00	1,200.00	0.00	2011
5 - Correlate compression block size with initial data set	Analysis / simulation: 22 alternatives	(none)	Definition and construction of data blocks Definition of construction of data blocks Methods to characterize non-uniform data	375.00	3,356.14	3,750.00	2011

1107 – MS .NET Development Issues

Project Details:

Scientific or Technological Objectives:

Measurement	Current Performance	Objective	Has results?
Responsiveness (startup time) (seconds)	1	0.1	No
Reduce the number of deadlocks (#/day)	2	0	No
Reduce memory footprint (MB)	1000	900	No

[A SOFTWARE INDUSTRY EXAMPLE PROVIDED BY AUTHOR]

We would like to improve the Performance and Scalability:

A number of application characteristics can affect performance and scalability including:
timing issues (deadlocks, races),
resource management (memory footprint, garbage collection, native resources such as window & GDI handles),
exception handling,
thread synchronization,
responsiveness (startup time, UI load time, response time & refresh rate in web applications, etc)
and ease of code maintenance.

Technology or Knowledge Base Level:

Benchmarking methods & sources for citings:

Benchmark Method/Source	Measurement	Explanatory notes
Internet searches	10 sites / articles	Blogs and articles on common errors
Similar prior in-house technologies	1 products / processes	We would like to improve our current software which is written with MS.NET
Potential components	10 products	Microsoft products

The Microsoft .NET Framework is a software component that can be added to or is included with Microsoft Windows operating system. It provides a large body of pre-coded solutions to common program requirements, and manages the execution of programs written specifically for the framework. The .NET Framework is a key Microsoft offering, and is intended to be used by most new applications created for the Windows platform.

Field of Science/Technology:

The Field of Science has not been identified.

Project Details:

Intended Results:	Improve existing materials, devices, or products
Work locations:	Commercial Facility
Key Employees:	None.
Evidence types:	Progress reports, minutes of project meetings; Records of resources allocated to the project, time sheets; Project records, laboratory notebooks

Scientific or Technological Advancement:

Uncertainty #1: architecture optimization METHODS

Key challenges:

While the .NET Framework can help with some of these issues by managing certain aspects (i.e. garbage collection,

exception handling) automatically, this managed environment tends to add an 'overhead', requiring more system resources than functionally similar applications that access machine resources more directly.

The most significant underlying key variables are:

Optimal thread synchronization (unresolved), Optimal execption handeling method (unresolved), Inter-operability (codes, databases, etc.) (unresolved), Optimal memory management (unresolved)

Activity #1-1: Potential SRED Activities (Fiscal Year 2011)

Methods of experimentation:

Method	Experimentation Performed
Process trials:	100 runs / samples Integration of Unmanaged Code: The .NET Framework provides the means to access the functionality that is implemented in programs that execute outside the .NET environment, but certain integration issues may still need to be addressed. Aside from issues of interoperability, additional performance issues and realted experimentation can be caused by: - late binding and reflection (this is a tradeoff - late binding results in code that is more flexible and adaptable, but at the expense of type safety, runtime performance and scalability) - unnecessary thread switching from failure to match threading models between the calling thread and the COM object - unicode (used by .NET's Common Runtime Language) to ANSI (used by many legacy components) string conversion

Results:

No results have been recorded for this Activity.

Conclusion:

No conclusion has been recorded for this activity.

Documentation:

- Offline Documents: .NET

Key Criteria Summary

Sample Software Industry

1107 - MS .NET development issues

Benchmarks: Internet searches: 10 sites / articles
Similar prior in-house technologies: 1 products /
Potential components: 10 products

Objectives: Responsiveness (startup time) : 0.1 seconds
Reduce the number of deadlocks: 0 #/day
Reduce memory footprint: 900 MB

Uncertainty: 1 - architecture optimization METHODS

Key Variables: Inter-operability (codes, databases, etc.), Optimal execption handeling method, Optimal memory management, Optimal thread synchronization

Activity	Testing Methods	Results - % of Objective	Variables Concluded	Hours	Materials $	Subcontractor $	Fiscal Year
1 - Potential SRED Activities	Process trials: 100 runs / samples	(none)	(none)	0.00	0.00	0.00	2011