



Government Agency Information Transaction System

The Background

This is a system of over 30 million lines of code maintained by a software staff of 400. The databases of the system are updated every two years. An assessment of the Year 2000 problem conducted several years ago revealed that 20% of the LOC would be affected. The owning agency decided that the problem solution would need to be transparent to the thousands of system users who use the system daily to deliver vital services to millions of citizens (the system generates between one and two million account payments per day). The selected strategy is conversion software injection.

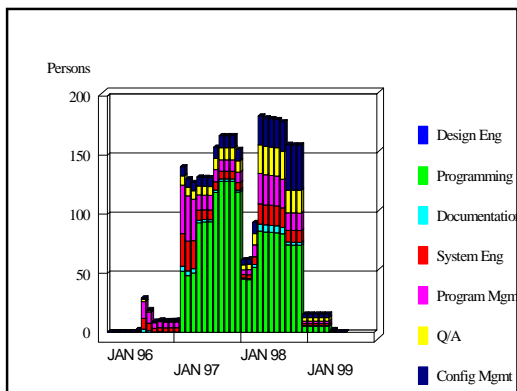
This is accomplished by:

- Designing and implementing the conversion code.
- Modifying the 20% of the affected code to interface with the conversion code.
- Integrating both of the above with the unmodified code and conducting system test.

The PRICE™ Software Model was used to assess the staffing profile required to complete the task. The estimated effort and time is 3025 labour months over 42 months with an average IT staff size of 75 and a peak of nearly 150. Started in June 1996, the task would be completed just in time to meet the close of the century. In order to complete the fix by January 1999, thereby allowing a year of test use, an average additional staffing of 50 (to 125) and a longer and higher peak (of nearly 200 for 18 to 21 months) would be needed. The effort in this case would increase by an estimated 26% to 3796 labour months. This agency is fortunate to have the necessary resources to meet either approach to fixing their Year 2000 problem.

The summarised results from PRICE S™ are shown below:

ALL Groups Staffing Profile



Gov't Agency Injection (JUN 96 - MAY 99)

	Gov't. Agency Injection						System	TOTAL
	Design	Pgming	Data	S/PM	Q/A	Config		
Sys Concept	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Sys/SW Req	0.0	0.0	4.0	38.2	2.5	2.5	47.3	47.3
SW Reqment	0.0	0.0	2.1	44.7	3.8	3.8	54.3	54.3
Prelim Des	0.0	63.0	5.7	91.1	10.4	10.4	186.5	186.5
Detail Des	0.0	103.5	8.6	136.6	15.6	15.6	279.8	279.8
Code/Test	0.0	997.3	16.5	124.8	79.9	79.9	1298.4	1298.4
CSCI Test	0.0	440.2	31.2	219.9	124.7	124.7	940.7	940.7
Sys Test	0.0	229.5	7.3	77.1	59.7	119.3	492.9	492.9
Oper TE	0.0	27.3	3.6	19.1	15.5	16.4	81.9	81.9
Sub-Total	0.0	1866.7	79.0	751.6	311.9	372.5	3381.8	
Sys Integ	0.0	243.3	8.6	105.4	28.3	28.3	414.1	414.1
Sub-Total	0.0	2110.0	87.7	857.0	340.3	400.8	3795.8	
Purchased Costs								0.0
TOTAL								3795.8

The case study analysis performed with PRICE S™ is described in detail overleaf. Background information about this case was obtained from the September 15, 1996 issue of CIO magazine (Pp. 53 - 64). The parametric modelling of each case consists of common elements to all Year 2000 problems and case specific elements. The common elements are discussed first.

If you would like more information about the Year 2000 problem, you can access the home page of Year 2000 at: www.year2000.com. If you need help, call us (01256 760012).

Common Elements

Specification Level:

Cases are treated as a high reliability, commercial proprietary software product (PLTFM = .9).

Management Complexity:

Cases are treated as though the software fix will be done in one location (CPLXM = 1). This is not a strongly supported assumption, since one of the cases will be fixed in part by using third party contractors in another country and at least one other case referred to use of external resources. This is an area where greater differentiation among the cases could be analysed.

Utilisation:

No effect considered (UTIL = .5).

Language:

COBOL used for all. This is somewhat of an oversimplification, since there will be some other language types in each system. However, COBOL is the dominant language. Additionally, previous analyses of commercial IS systems like these reveal that the effort required to account for these other languages is unjustified since it does not significantly alter the analysis results.

Productivity Factor:

A factor of 12 is used throughout. This is the highest reference value of the PRICE S™ Productivity Table. In addition, this value has been validated for use in calibrations of other commercial systems. The value must always be kept in the following context, however - it is on the high side of our observed project performances and its appropriateness to COBOL programmers in the late 1990s is not obvious. For these reasons alone, all results should be interpreted as optimistic.

Fraction of Non-executable LOC:

50% is used throughout (FRAC = .5). This is based on the high incidence of data sections of COBOL applications. 50% is conservative, as we have noted values well above this for similar software systems.

Development Environment Complexity:

Nominal values are used for personnel skill level, product familiarity, software tools, and complicating factors (CPLX = 1).

Hardware Complexity:

No hardware design or usage complications are considered (CPLX2 = 1).

Application:

Cases are treated as software composed of data

storage and retrieval modules (50%), Online Communications modules (20%), String

Manipulation modules (20%), and Mathematical/Statistical modules (10%). This equates to an APPL of 3.83.

Functions & Tasks:

The following functions (with percent of PRICE S™ defaults) are included: Programming (100%), Documentation (20%), Systems Engineering/Program Management (70%), Quality Assurance (50%), and Configuration Management (50%). The following tasks are included: System Software Requirements Analysis, Software Requirements Analysis, Preliminary Design, Detailed Design, Code and Unit Test, Module Test, System Test, and Operational Test and Evaluation. For the rationale of these values, please see the PRICE Systems™ Metric File documenting commercial software settings for use with PRICE S™.

Case Specific Elements

Internal & External Software Integration:

The software integration within and across each modelled item (system, subsystems, modules) was treated as closely coupled and as being performed by experienced software developers (INTEGE & INTEGI = .5). This assessment is based on the need to thoroughly test each item because of new software injection.

Project Start:

The project start is January, 1997 for every modelled item. Sufficient staffing allows for all subsystems to be started at the same time.

Conversion Algorithm:

This was modelled as a totally new 5000 LOC application to reformat dates from 6 to 8 digits and assure correct interpretation of all date calculations.

Modified & Redesigned 20%:

20% of the 30 million LOC, or 6 million LOC require repair. This is modelled as a series of subsystem developments, each working on 120K LOC. One of these systems is used to account for all software design, which is limited to 20% of the 120K LOC. The remaining 50 subsystem models require only recoding to the design. CMULT is used to facilitate the modelling and reduce the amount of data input.

Unmodified Code:

The 80 % of the system that is not unchanged is modelled as 50 identical subsystems of 480K LOC, which are tested as a system when the repaired code and algorithm are injected.