Year 2000 Case Study

Consumer Products Company Information System

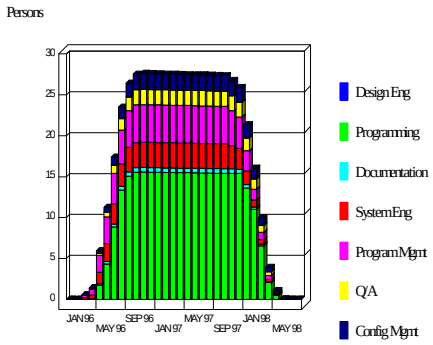
The Background

For the past 5 years, this company has designed all new and replacement software to the 8 digit date convention required for Year 2000 operation. Thus, the replacement strategy is a part of this Year 2000 problem attack. Unfortunately, some 6000 modules of heritage systems will remain in use and need conversion. The fix for these has only recently begun. The strategy being employed is system upgrade on a priority basis - business conditions are such that a limited staff (with outside contractors) is assigned to the project. Because of the relatively late start and staffing constraint, there is some doubt that this company will convert all software systems in time.

The PRICE™ Software Model was used to assess the staffing profile required to complete the task. With an average staffing level of approximately 21, and allowing for a peak of 28, it is estimated that conversion can be complete in 28 months, with an effort expenditure of 550 labour months. This effort includes test of the subsystems only, and does not allow for overall system test. By starting now, this company can implement repair in time to operationally test the revised system during 1999.

The summarised results from PRICE S™ are shown below:

ALL Groups Staffing Profile



Consumer Products IS (MAR 96 - JUN 98)

Acquisition Costs / System

Consumer Products IS System
Costs in Person Months

	Design	Pgming	Data	S/PM	Q/A	Config	TOTAL
Sys Concept	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Sys/SW Reqt	0.0	0.0	0.4	3.8	0.3	0.3	4.7
SW Reqment	0.0	0.0	0.6	20.9	1.1	1.1	23.6
Prelim Des	0.0	25.2	1.8	32.1	3.3	3.3	65.7
Detail Des	0.0	37.8	2.7	48.2	5.0	5.0	98.6
Code/Test	0.0	213.6	4.2	33.6	19.4	19.4	290.1
CSCI Test	0.0	23.8	1.1	10.6	4.8	4.8	45.1
Sys Test	0.0	6.3	0.2	2.1	1.6	3.3	13.5
Oper TE	0.0	2.7	0.4	1.9	1.5	1.6	8.1
Sub-Total	0.0	309.3	11.3	153.2	37.0	38.7	549.4
Sys Integ	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Sub-Total	0.0	309.3	11.3	153.2	37.0	38.7	549.4
Purchased Costs							0.0
TOTAL							549.4

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.

PRICE Systems • Lockheed Martin (UK) Ltd • 18, rue Horace Vernet • 92136 Issy les-Moulineaux • Cedex • FRANCE
Telephone [1] 47 65 76 33 • Facsimile [1] 47 65 76 92

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:
There is no integration or testing conducted among the subsystems of this system. The module integration within each subsystem is considered to be to closely coupled interfaces (INTEGI = .5), while that of the subsystem integration and test is to loosely coupled interfaces (INTEGE = .3). An experienced team is considered.
Project Start:
The project start is April, 1996, which is when the first subsystem modification began. Other subsystem modifications start in a staggered fashion from April 1996 through to November 1997.
Subsystem Model:
This project is modelled as 100 subsystems (CMULT is used to expedite data entry and analysis) requiring modification to a small amount of software and integration and test with that which is not modified. The LOC estimate is 1500 LOC per module, which when applied to the 6000 modules identified as needing repair, results in 9 million LOC.
Modified Code:
Based on other Year 2000 analyses, a 5% change factor was applied to estimate the number of LOC needing modification. The higher percentage applied is due to a higher anticipated date occurrence incidence in the heritage code that was not replaced by previous migration activities; there is a tendency to not replace if the code that requires greater change. This results in 450K LOC for the system, or 4500 LOC per subsystem. Each changed LOC was considered to require new design as well as new coding.
Unmodified Code:
This is the remaining 95%, or 85500 LOC per subsystem that is integrated with the modified code and tested at the subsystem level.