

Presentation # 5005
The YEAR 2000 from a Business Perspective

by John J. Whitehouse, Director
The Progeni Corporation
31 Holcomb Bridge Road
Norcross, Georgia 30071
770.840.7550

The *Year 2000 Problem* is not what you think it is..... You have no doubt seen articles about the Millennium Opportunity. You've read about it and understand that there are possible problems in the way computers will look at the "zero-zero" in the date field and not know whether the date is 1900 or 2000. Maybe you've done a detailed assessment of the hardware and software your company depends on and really know what kind of problem you have. But, more likely, you're in the high percentage of executives who really don't know if they have a problem. This is what the *Year 2000 Problem really is all about* - A Lack of Awareness and Action.

There have been many articles that talk about the gloom and doom computers will face as we end 1999, or even before, as our computer programs look forward to the year 2000. It is clear that all companies will have some kind of an opportunity to deal with as a result of the millennium shift. It is just as certain, today in early 1997, that many companies have not accepted this fact. They either think they don't have a problem, or because they're too busy now or don't have the monetary resources, haven't taken the time to look at the issue yet.

Many executives haven't really taken the problem seriously. But even more distressing is the fact that many Chief Information Officers believe that the "zero-zero" date problem in programs won't affect their firms. This is not surprising to me since I present Executive Level Awareness Seminars around the country and I find a great misunderstanding in existence as to what must be done to determine the magnitude of the problem in a particular enterprise..... If you do not have a detailed inventory of the hardware and software components that make up the mission critical systems at your company; and if you have not tested each one, you don't know if you have a problem! If you assume you don't have a problem because someone, either from your company or from a vendor told you that you don't; you are probably

Year 2000
5005-1

in deep trouble. The cost to determine the magnitude of your problem may affect your bottom-line, but the alternative is to bet your business on what someone told

you without really researching the problem. Are you willing to take the risk? **I strongly recommend that you have evidence that you have done your homework and completed the due diligence process.**

The “zero-zero” problem in computer programs is only part of the potential malfunction in our systems. Many hardware components that have computer chips in them will have problems dealing with the new millennium. PBX’s or telephone switchboards, bar-code scanners, GPS’s or Global Positioning Systems, routers on networks and the BIOS or basic input/output systems of PC’s have all shown problems when tested. Since the year 2000 is the first century since 1600 to be a leap year, I wonder how many digital calendar systems will show February 29, 2000. As you can see, our exposure to the problem is greater than most of us would assume. The Gartner Group predicts that over six hundred billion dollars will be spent worldwide to resolve the issue. To put this in perspective, World War II had an international price tag of three hundred and fifty billion dollars. At congressional hearings in May, the U.S. Federal Government estimated its cost to be about thirty billion dollars. A bill has been introduced in the Senate to create a National Commission to do a detailed study on the magnitude of the Federal problem. Many huge U.S. corporations have already spent hundreds of millions of dollars apiece to address this problem.

Are all of these organizations wasting their time and resources or do they know something that some of us don’t??? **Soon many of us will learn more about this, because insurance companies providing Directors and Officers Liability Insurance or Business Interruption Insurance have said that in 1997 they will begin refusing to insure companies that don’t have detailed Year 2000 Compliance Plans.**

The first step toward awareness as an executive is accepting the fact that your enterprise has a potential exposure to the Year 2000 Problem. The second step is trying to learn, from a business perspective, what the problem is all about and how it might affect you. Once you have a general understanding of the issues, it is critical that this knowledge be communicated to all executives within the enterprise. Keep in mind that this is not just an Information Systems problem, it is a

Year 2000
5005-2

general business problem that will affect most all divisions of your company. There are major financial issues, legal issues, public relations issues, personnel issues, and potential operations issues in every division that uses a PC, or a spread sheet, or word processing, or an intra-departmental database program. If you are not the CEO or CFO, it is imperative that they be made aware

of the potential problem. If you're afraid to discuss this with the CEO for fear that the messenger will be blamed, buy a consultant to be your messenger to the CEO. Your "Executive Committee" should then meet and discuss this and come to a comfortable understanding of the issues. If there are executives that still feel that this is a trivial matter, they must be won-over to the cause. Executive commitment is critical to success in this project.

Process control is the most important factor in this entire effort. Once there is overall management understanding, a full-time project manager must be selected at least for the duration of the impact study. When making the selection, it is very important to take into account that this may well be the largest and most critical project the company will undertake over the next few years and the candidates for the position should possess the appropriate experience and skills.

The project leader, once selected and commissioned, must with top management support, organize a company-wide Year 2000 Task Force and develop a Task Force Charter. It would be the responsibility of this group to complete the Year 2000 compliance assessment for all hardware and software and develop an impact analysis statement for top management. Their recommendation should include the alternative courses of action and a proposed high level plan of action.

To begin the actual Assessment and Impact Analysis Process, we need a detailed inventory of all systems. This sounds simple but believe me, it's not. The Information Services Division may have a list of the systems they process but this is only a start on what we need. To get the required systems inventory we need a bottom-up analysis which looks at the information used by each division of the company and where it comes from. Also what information comes into these systems and what information goes out of these systems to our third party trading partners. What we are really looking for is a detailed list of all computer systems that provide any information needed to keep our enterprise running smoothly. We must know what they are, all their

Year 2000
5005-3

hardware and software components, and their relative criticality to the strategic goals of the company. Every component is important including: all programs, connecting software or middleware, called routines, and network hardware and software. If you remember, I mentioned before that routers have a problem and so might the microprocessors in the satellite receivers. It's critical that we identify all components or we may end up with broken links we haven't tested.

At this point, even a relatively small company will have a very long list. The components need to be categorized by those that have been developed or built internally and those that have been purchased or leased from third parties. This systems inventory should be stored with all its attributes in an electronic data dictionary or repository where it can be sorted, referenced, and updated as we proceed through the process. For those components that we have purchased or leased it is critical that we send an effective and legally correct letter to the vendor asking: when their component will be “Year 2000 Compliant”, what versions of other software or hardware are prerequisites, how much it will cost, and when the compliant version will be available for client use and testing. The internally developed software must be somehow analyzed to determine if there are date occurrences that are non-compliant, their interfaces, and their complexity. To do this manually is very costly, laborious, error-prone, and in some cases unfeasible. If you have hundreds of thousands or millions of lines of COBOL code that form the backbone of your information systems, then an automated alternative is much more practical and in most cases less costly. Fortunately there are automated tools and companies experienced in the process that are available to help in this analysis.

These automated tools scan and parse the code looking for date related data elements and their references. They are much more effective than a manual process because they are consistent, persistent and much faster. If you elect to use such an automated tool for analysis; your code, copy books, etc. for an entire system are fed into the tool and using the parsing engine and the scanning routines, a database of all data elements from all programs and their interfaces is built. This database is then subjected to the date identification routines which takes into account specific information that is unique to your organization and any known coding conventions

Year 2000
5005-4

which were used in system development. These routines not only identify the date locations but how they are used, their interfaces and the relative difficulty to make the necessary changes. The resulting reports summarize, at the program level, the severity of the problem and the estimated effort that will be required to bring the systems into compliance. This type of information is invaluable in completing an impact analysis and evaluating the alternative courses of action in bringing the systems into compliance.

A key metric affecting Year 2000 code conversion is Date Complexity. Measuring this value for your programs requires a detailed analysis. Date Complexity

reflects the number of lines of code with date references, the number of interfaces (to databases, files, screens, reports, other programs, etc.) with date content, and a program's cyclometric complexity. Using an automated scanning and parsing program with this level of sophistication can show date complexity by program and compare this to lines of code in each program. This is very relevant in determining the effort required and the programmer competency required to make the programs Year 2000 compliant. It's interesting to observe that, although we might think that the number of lines of code is a reliable indicator of complexity, often our derived complexity metric shows this to be untrue.

Estimating the Year 2000 date code conversion costs is critical to arriving at a realistic impact analysis. An automated system with the appropriate logic can estimate the costs for the different approaches to conversion using the unique characteristics of the code analyzed. It takes into consideration the data derived from the automated analysis and uses the complexity metrics. The cost of either a manual or an automation assisted solution can be estimated when the programmer cost per hour is known. When developing the programmer cost per hour, it is very helpful to have the information previously mentioned that accurately showed the competency level requirements.

When you have completed the analysis of all of your internally developed code and received the replies from the vendors of your purchased or leased code and hardware, you are in a position to say you truly know if you have a Year 2000 problem and what and when you're going to do something about it. At this point you have completed Phase I of the Compliance Process which is the Awareness/Analysis

Year 2000
5005-5

Phase and in my opinion the *Real Year 2000 Problem*. But, believe it or not this is only 30% of the entire process to fix the problem. It's estimated that modifying the code is another 20% and testing and implementing the compliant code still takes about 50% of the time and effort. If you've gotten this far, you understand the problem, you see the light at the end of the tunnel; there's hope because you've taken the Mystery out of the Year 2000.

Now you know where most of your problems are, you know which ones will be the most difficult to fix, and you know which ones must be fixed first. The key to success at this point is using the repository of information you built during the assessment phase to guide and control the modification phase of the process. If you used a

sophisticated automated tool like we talked about earlier, your repository should contain the data fields that deal with date along with their attributes, relationships, and interfaces. This information was more than adequate to provide an accurate assessment of the problem, but to gain an high degree of automated modification, it is important to validate this data using an automated browser that can show in multiple windows on a screen the data element involved, its use in the program, and its relationships throughout the system. Once you have validated the date occurrences and your repository is complete, you have an invaluable reference point for your total modification and testing process.

It is impossible to freeze an entire system while the Year 2000 Compliance Process takes place. Using our data relationship repository and the knowledge we gained in our assessment as to which occurrences will cause a problem the earliest, we can now divide the systems up into “chunks” of programs that will be modified and tested as a group. At this point it is critical that we plan the process for when our system is in a state of transition. As one group of programs is modified to have four position century dates, yet unmodified programs that they communicate with may still have two position century dates, and our data base itself may well have data elements that programs in many different groups use and update. As you can see, this becomes a very complex problem if your modification process is not automated and based on the information contained in our reference repository.

Most modification solutions use an external “data bridging” procedure that requires a program be called to do table lookup and data transformation so the

Year 2000
5005-6

interface with non-compliant programs and data can be accomplished. A better approach is to temporarily include in the program being modified, a flexible interface macro that can pick up from our reference repository at generation time, the system program and data interfaces and internally match the external interface at execution time. However we handle these interfaces, transition must be accounted for as we plan our modification, testing and implementation.

Other factors to be considered in planning are the availability of personnel and hardware resources for modification and testing. This project plan is probably the most complex your organization has considered for a long time. Don’t underestimate the effort and skill level of the project manager selected. Timing of your own modifications, availability of compliant vendor products including operating systems, middleware, and

communication utilities, and the replacement of any non-compliant hardware are but a few factors that will determine your plan.

Since the system has been logically divided into manageable “chunks” of code, a selected “chunk” can now be run through the automated modification process using the conversion parameters derived from the data reference and interface repository. Experience has shown that a high percentage of successful automation-assisted modification is achievable using this approach and the Progeni tools. The areas of code flagged as requiring manual review can then be looked at and corrected. If the reference repository was properly validated earlier in the process, the manual portion to be converted should be less than ten percent.

Now begins the most difficult and time consuming part of the Year 2000 Compliance Process, the testing of the modified code and its implementation into production. Because the changes have been so pervasive and interfaces are three dimensional with the additional time variable, most experts agree that this will probably take about fifty percent of your time and effort. If your planning process was thorough and complete then this phase may not be as difficult as might be expected. Some key critical success factors include: the availability of a proper test hardware and test software environment, the timing of vendor supplied compliant software, the interface of modified and yet-to-be-modified data

Year 2000
5005-7

elements, and the construction of comprehensive test scenarios.

No vendor can successfully provide a turn-key solution in this phase. Your users and application programmers having some familiarity with the system being tested are essential to success in the test phase. When developing test scenarios, in addition to your people knowledgeable on the system, the reference repository is again invaluable. An automated test tool is also important. Many tools are available in the marketplace, you may already have one in use for testing your normal program maintenance today. To properly test for Year 2000 compliance and potential disruption of existing processes, a tool is required that can record your test scenario, replay the data input into your baseline system and into the system with the modified programs, and can then record the printed, screen, and electronic output and data changes resulting from each scenario. The Progeni 3 R's tool set does all of this and also provides a comparison process that can filter the expected

changes and report on those that are unexpected so they can be analyzed to determine if a problem exists. If a problem is present, the code must be fixed and the scenario must be re-tested. Before the “chunk” of code is ready for implementation into production it must be tested not only in the Year 2000 time environment but also in the current time environment with the current hardware and software.

When the code has been tested and you’re ready for implementation into production the process isn’t finished yet. Make sure that you have prepared for the inevitable latent problems that will occur. Make sure you have prepared for possible rollback of all new components in the event of some unforeseen disaster. Have your disaster recovery plans updated for the new hardware and software. Change management is never more important than during this process. Involvement of the business areas in the corporation is very important during the modification and testing phase but it is critical during the implementation phase.

The entire millennium opportunity is an exercise in risk management. The problem is immense and difficult to comprehend; the solution is highly complex, almost unmanageable and very costly; and the risk is nothing less than the survival of your business. There is probably insufficient time to correct every possible problem. Success will be based on getting started as soon as possible, committing yourself and your

Year 2000
5005-8

people to the process, using proven solutions, consultants and vendors experienced with the process, understanding the risk, evaluating tradeoffs, and being decisive. Those enterprises that complete assessment, code modification, testing, and implementation by early 1999 will drastically improve their chances for success. Your efforts now could turn the Year of the Dragon into a Year of Opportunity.

John J. Whitehouse is President of Changewise, Inc., a Director of the Progeni Corporation, Adjunct Professor at the University of North Florida, a Year 2000 keynote speaker, and is a member of several Corporate IS Advisory Boards. Mr. Whitehouse has had over 25 years of management experience in the IS industry. He specializes in executive level consulting on Year 2000 issues. He can be contacted at www.progeni.com on the Internet.

Year 2000
5005-9