STRUCTURED SOLUTIONS

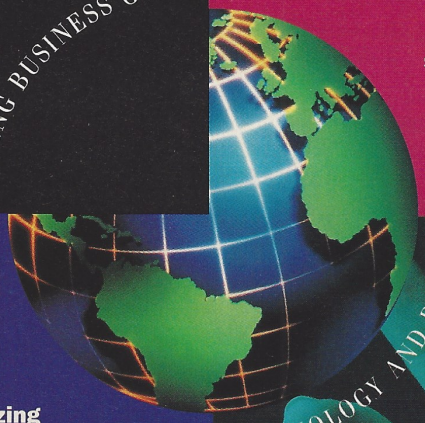
Structured Solutions offers a single methodologist to design a comprehensive approach, analyze training needs, monitor the development process and consult on systems development issues.

Structured Solutions also offers a full suite of products to support the development process including project administration, project estimating and information gathering - all in an automated, multi-tasking environment.

By uniting all of these resources - methodology, tools, training and consulting, Structured Solutions enables the development of world-class business systems.

If creating world-class business systems is important to you, call Structured Solutions today at 1-800-325-1087.

UNITING BUSINESS GOALS



TECHNOLOGY AND PEOPLE

Structured Solutions enables the development of world-class business systems while maximizing your internal resources. The process management strategies used integrate methodologies, tools, training and consulting into a comprehensive approach.

This approach provides strategic direction, accelerates development and assures quality through AD/*Method*®, a full life-cycle development methodology. The methodology offers you the flexibility of a variety of routes through the development process, most notable - Client/Server. The methodology is fully automated, easy to customize and provides a structured approach to developing business systems.

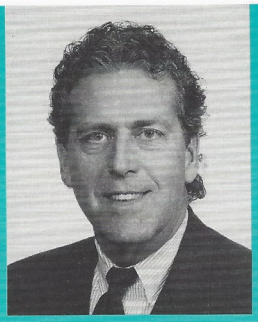
ENABLING WORLD-CLASS BUSINESS SYSTEMS

STRUCTURED SOLUTIONS

400 Interstate North Parkway
Suite 800, Atlanta, Georgia 30339
(404) 618-7900, FAX: 618-7909

CIRCLE #24 ON READER SERVICE CARD

From the President



Robert V. Connolly
President, Structured Solutions, Inc.

All too often our industry becomes enamored with front page industry articles espousing the virtues of the "next technology." If this next technology benefits the end-user in providing world-class information, it has tremendous value. Client/Server brings such value to the end-user. The essence of this technology is to provide accurate information, responsively, at the right place, right time, right format and right cost.

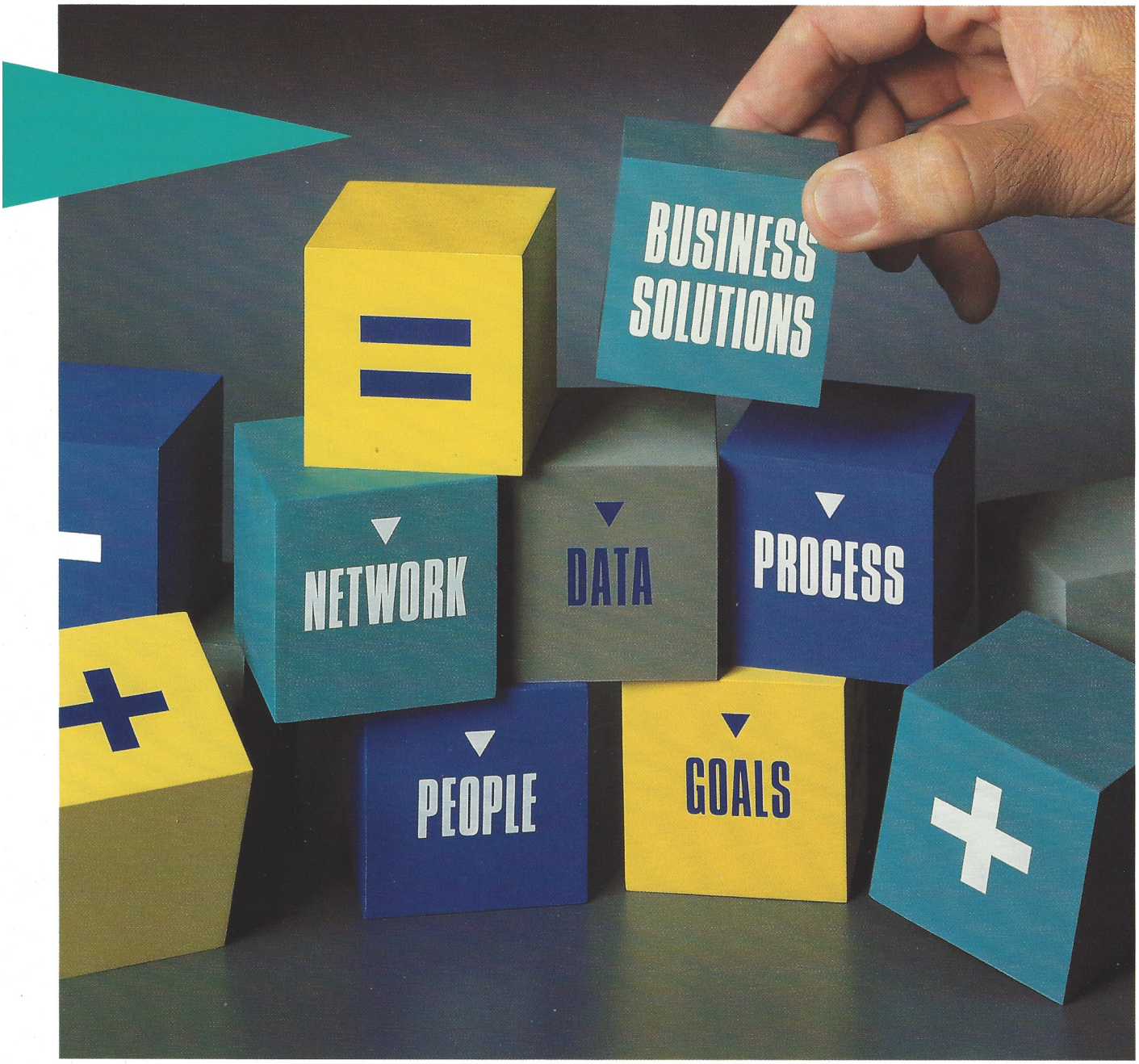
How does client/server achieve these benefits? By placing the right information close at hand to the end-user, in a cost effective manner. Easier said than done. To effect this simple rule of client/server, data, process and presentation, elements must be rationally distributed, centralized or decentralized based on an assortment of rules governing their proper placement. For instance, say 50 user locations need access to the same "master" data that only one location updates, and these remote locations need it "fresh" as of 24 hours. Then we could rationally "replicate" the master data to all 50 locations with updates distributed every 24 hours. Also, the processing required to "update" the master data need only reside at the location performing the update. The processing and presentation format required to access this master information could be placed at the 50 locations. This type of distribution benefits the end-user in that the data, processing and human interface is at their respective location and is responsive, accurate and probably more cost effective. There are about 18 different scenarios like the former example that could be applied to client/server data, process and presentation distribution. Determining this distribution is an example of what role a methodology must play.

The essence of this technology is to provide accurate information, responsively, at the right place, right time, right format and right cost.

A number of companies I have visited to demonstrate our own AD/*Method* for Client/Server methodology, have said to me they want to eliminate the mainframe - that is the primary reason for turning to client/server. I don't believe that is the proper premise. Remember my earlier statement, "The essence of this technology is to provide accurate information, responsively, at the right place, right time, right format and right cost". This may mean that the mainframe as a "super-server" should in fact, be part of the solution. If the "getting rid of the mainframe" statement was initiated because response time was poor, user interfaces were not friendly, enhancements took too long, etc., then client/server to the rescue for sure. The mainframe may not be dead since it could play a role as the "super-server" helping with proper physical data and process distribution to hundreds of locations with smaller server configurations. In other words, it could become a wonderful "traffic cop" for housing and distributing shared data and associated processes.

It is apparent to many, including myself, that the "front pages" promoting client/server are right-on. This bit of technology, accompanied by a methodologically sound life-cycle, will be the future of computing. Not necessarily for the technology itself, but for the fact that it is the end-user - the business, that reaps the greatest benefits of all.

Robert V. Connolly
President, Structured Solutions, Inc.



AD/Method™

Accelerated Development Methodology

for Building Information Systems

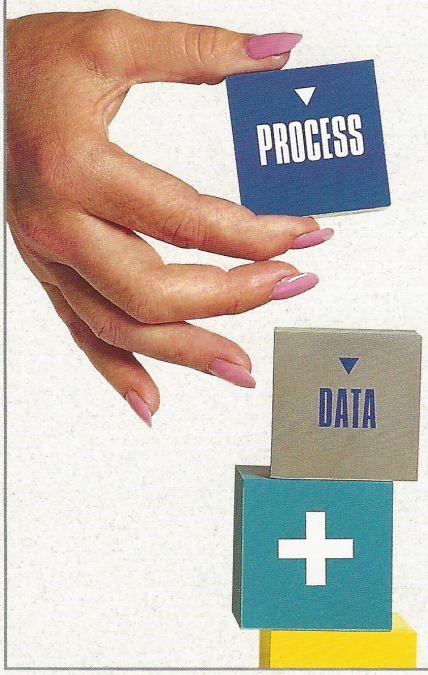
STRUCTURED SOLUTIONS

AD/*Method*™

AD/*Method* is an accelerated development methodology for building information systems. This methodology uses a series of building blocks to create solid business solutions. It is adaptable, practical, and tied to CASE.

AD/*Method* was born out of frustration with applications that are late, inadequate to user needs, difficult to maintain, and in some instances, never delivered.

This methodology seeks to bring standardization and discipline to the system development process. It uses rigorous, scientific approaches while relying on business goals and user feedback to insure the system meets the needs of the people who depend on it.



AD/*Method* helps manage the frustrations of the ever-changing needs of business. Changes in business requirements are reflected by following AD/*Method* procedures to reconstruct an updated version of the system.

AD/*Method* increases productivity while maintaining quality through the integrated use of tools, techniques, and models.

This methodology simply provides the building blocks for constructing useful, maintainable systems.

People + Goals
+ Data + Process
+ Network
=
Business Solutions

Solid Building Blocks

AD/*Method* gives you the solid building blocks for successful system development. They focus on the needs of people, the goals of the organization, the data to be managed, the processes for managing, and the hardware and communication network. They are the foundation for providing real business solutions.

AD/*Method* provides the tools and techniques to conceive, prototype, blueprint, build, test, and deliver working systems, in an accelerated fashion. This methodology attunes the development process to the needs of people as well as the application's business objectives.

Sound Foundation

As business applications become more complex, they cannot be built effectively without proper engineering. Engineering sciences have long been characterized by:

- **standardization** - a common methodology
- **discipline** - a system of rules governing conduct
- **rigor** - a strict level of precision
- **tools** - to enable maximum productivity
- **metrics** - for rigid analysis and comparison
- **feedback** - for evaluation and improvement

Software engineering is no different! AD/*Method* provides this foundation.

Why AD/*Method*?

Usability is critical to a methodology's acceptance. To insure that acceptance, both the automated and hardcopy versions of AD/*Method* may be tailored to fit your organization's culture.

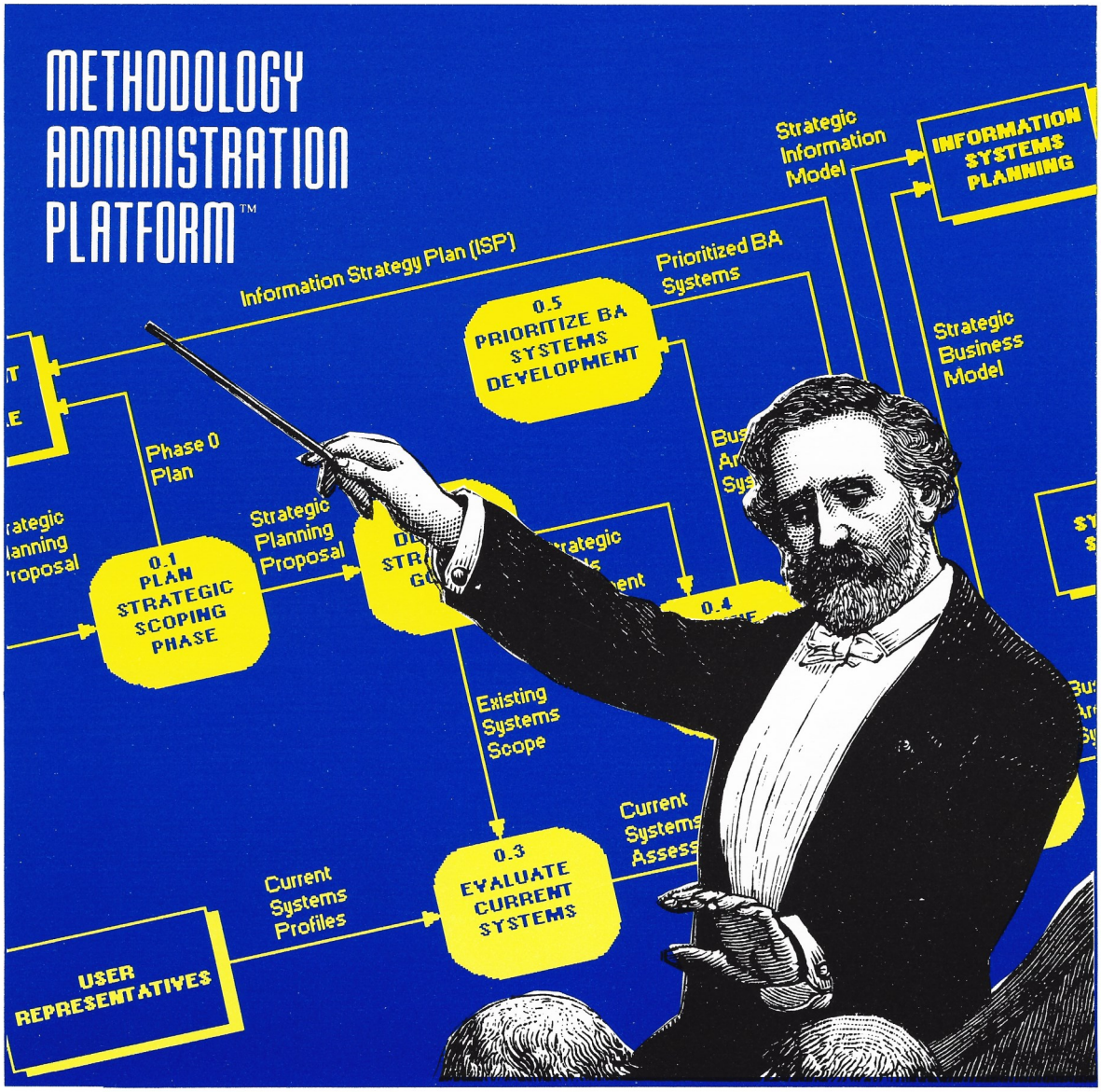
AD/*Method* incorporates project management software and many different project "templates". These templates provide multiple project management plans that show timelines and the parallel/sequential relationships of all activities within the life cycle.

Structured Solutions, Inc. supports this methodology by a complete curriculum of education. This education teaches software engineering with CASE and within the AD/*Method* life cycle.

In short, this methodology offers you the building blocks for success.

STRUCTURED SOLUTIONS

MAP/ADMINISTRATOR™



Orchestrating Your Workflow by Integrating Methodology & Tools into an Automated Administrative Platform

INSIGHTS

July 1994

A Biannual Update from Structured Solutions, Inc.

Sprint's High Standards Lead To Success with AD/Method for Client/Server™

Steve Sage, Methodology Consultant, Sprint

In today's fast-paced development environment, Sprint Corporation needed a client/server methodology that actually reduced development time and simplified the complex analysis process. Sprint had already developed client/server systems with the help of outside consultants but needed a simple, methodological solution.

The search was on for a methodology that included all of the components of client/server development in one manageable package. To guide them in their search, Sprint formed a team that established the criteria for a client/server methodology. This criteria included:

TOOL INDEPENDENCE:
The methodology needed to be tool independent. Several methodologies were available within a tool suite but few rigorous methodologies were available independent of a tool or platform.

ARCHITECTURAL INDEPENDENCE:
The methodology needed to be architecturally independent as Sprint has a large and diverse technical architecture. The methodology would have to be compatible with and take advantage of the existing architecture and allow Sprint to access legacy systems and databases, take advantage of diverse hardware and use internal expertise in network and communications protocol.

CULTURAL SENSITIVITY:
The methodology needed to be culturally sensitive to take advantage of the current development culture. With many hours and dollars invested in systems development methods and techniques training, the new methodology would have to blend well with the information engineering-based principles and industry recognized metrics which were already in use. The methodology had to be compatible with instilled development processes such as facilitated analysis sessions and milestone walk-throughs.

PROTOTYPING TECHNIQUES:
The methodology had to contain prototyping techniques which Sprint found to be a most valuable development technique. Sprint uses several types of prototyping throughout the development life-cycle for everything from discovering requirements to user verification.

ITERATIVE DEVELOPMENT:
The methodology had to take advantage of iterative development. Sprint was looking for a methodology which would clearly illustrate that components of systems development could be built modularly and simultaneously. A client/server methodology would need to demonstrate a 'parallel recursive' life-cycle that is manageable. Iteration could not be seen as rework.

After searching unsuccessfully for a methodology meeting these standards, Sprint decided to once again call on the expertise of the people at Structured Solutions. Sprint proposed a partnership

(con't on page 8)

IN THIS ISSUE

3 CLIENT/SERVER UPDATE

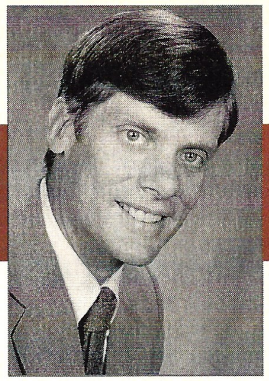
5 STRATEGIC VISIONS

6 TIPS & TECHNIQUES

7 '94 MASTRS CONFERENCE

CUSTOMER SPOTLIGHT

STRUCTURED SOLUTIONS



Charles F. Martin

NEW APPROACHES IMPROVING ACCESS TO 'BIG' METHODOLOGIES

SSI's AD/Method offers flexibility, industry-specific options

[Author's Note: The pendulum always swings back. We have been through a period during which methodology was unimportant. Any way that client/server applications could be implemented was acceptable. Now we are attacking larger, more critical applications, and linking them with reengineered business practices. Methodology again is a pacing item in allowing us to develop information technology effectively and quickly enough to support the business. Meanwhile, some fascinating new methodology concepts have come out with little fanfare. This column presents some alerts about and insights into the best of these new methodology ideas.]

As a manager of a large, new application development project, how would you like to have a team of successful project managers and team leaders advising you? They could guide you based on lessons they have learned from successes and failures. You would not have to learn all these lessons the hard way and get early answers to all of the following questions:

- How should you divide the project into discrete activities?
- In what order should you accomplish them?
- Who should be involved?
- What types of inspections and reviews will ensure quality?
- What results (deliverables) should come out of each activity?
- What techniques and tools will help the team perform the work efficiently?
- How much effort and time will the whole project require?
- In short, what should you use for the project methodology?

Charles Martin, president of Charles F. Martin Associates Inc. in Concord, Mass., is a software engineering methodologist who is constantly looking for improved methods and tools to help his clients be more effective with information technology. He can be reached at (508) 371-7011 or cmartin@world.std.com, and would appreciate your comments.

This kind of expert advice has been available since the early 1970s in the form of packaged methodologies. You have heard of them: Method/1 from Anderson Consulting, Chicago, Navigator from Ernst & Young, New York City, 4Front from Deloitte & Touche, Wilton, Conn., Information Engineering Methodology from James Martin & Company, Reston, Va. But you probably do not intend to get on the phone and order one to help support your project.

These "big methodologies" — full life-cycle recipes — have not proved to be the panaceas that people hoped in the 1970s and 1980s. Certainly they offered some excellent advice from people with real project experience. However, they were tough to assimilate into real software development shops. Packaged as bookshelves full of two-inch thick three-ring binders, they were rather imposing to individuals on the project. For example, to find out where in the process I should start entity-relationship modeling, I did not want to wade through a thousand pages of documentation. Moreover, it proved to be difficult to modify these materials for particular organizations and projects. I observed one large MIS organization who spent $100,000 on a methodology and then about three staff years to customize it for that company. After customizing, it was still difficult to use for individual projects.

Besides, all companies and all projects are not the same. Rapid prototyping development and implementation using commercially available software differ from classical lifecycles. True, the methodology vendors added multiple paths to their offerings and that helped. Even so, it was still hard to implement the methodologies in real development organizations. Interest in big methodologies waned in the late 1980s and early 1990s.

But now I am starting to see some intriguing new approaches to big methodology that make them much more accessible to software development organizations. Particularly impressive are the offerings of Structured Solutions Inc. (SSI) of Atlanta. Like Burger King, its approach allows you to

"Have it your way." Moreover, SSI's personal computer tools provide a far more effective methodology management and delivery mechanism than the old-fashioned bookshelf.

STRUCTURED SOLUTIONS

Structured Solutions is a consulting firm started in 1983 to help Fortune 500 IS departments improve performance with appropriate methodology. According to SSI president Bob Connolly, SSI emphasizes facilitated sessions and business event modeling. Its consulting assistance aims to make the client independent and a good reference for SSI. Tools became a primary mechanism for achieving that independence.

Here is how the approach works. SSI starts with a basic big methodology called AD/Method. The backbone of AD/Method is a set of six phases, starting with "System Scoping" and ending with "Distribution and Installation." "Reengineering and Maintenance" comes as a seventh phase after initial implementation. Although they differ from Information Engineering and Department of Defense life-cycle views, these phases are reasonable and intuitive. Each phase divides into activities and then steps. Steps list the inputs and outputs, personnel roles, and techniques involved. These elements give a solid methodology base but are not particularly exciting.

The really interesting part is how SSI packages AD/Method and allows the organization to customize the methodology. (See Fig. 1.) MAP/Administrator is a DOS/Windows software environment that allows the methodology administrator to customize the methodology to suit the software development organization. MAP/Viewer is another Windows tool that acts as the delivery mechanism for project participants. Both can operate over a LAN.

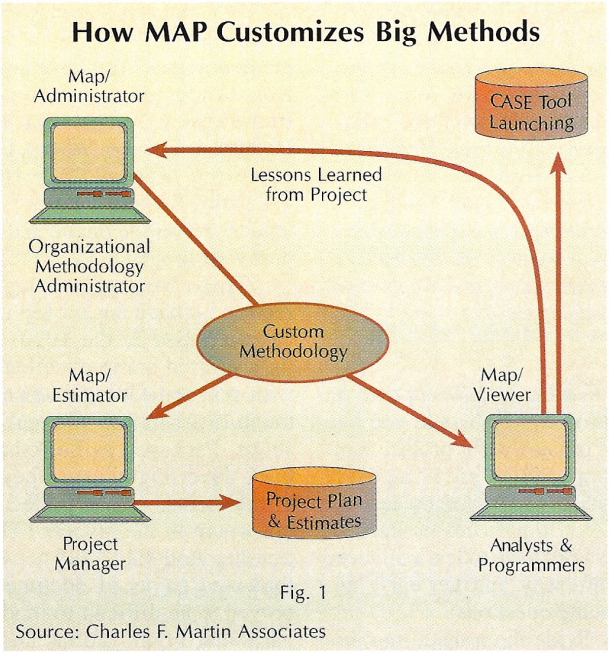


Fig. 1

Source: Charles F. Martin Associates

The combination of MAP/Administrator and MAP/Viewer solves the implementation problems associated with big methodologies. MAP/Viewer makes it easy to access concepts, based on the pictorial guide of the phase, activity, and step flow diagrams.

ROUTES THROUGH BIG METHODOLOGY

The basic AD/Method comes packaged as a large variety of project-type variants, called "Routes." For example, there are routes for time box development, rapid application development, and package development. Standard life cycle routes include constrained development, in which budget and/or schedule constraints are critical factors, and fast path, which simplifies steps to the maximum extent, in order to speed up development. For each route, AD/Viewer gives flow diagrams of the phases, activities, and steps. At the step level, there is a tool bar with command buttons to display a roles/techniques matrix, list of inputs, list of outputs, special instructions, and examples for that step. In addition, the methodology administrator can enable launching an application. For example, a step for data modeling could include a launch button for starting the CASE tool used for entity-relationship modeling. The methodology administrator can add or delete routes as appropriate for that organization.

This combination of MAP/Administrator and MAP/Viewer solves the implementation problems with Big Methodologies. MAP/Viewer makes it easy to access concepts, based on the pictorial guide of the phase, activity, and step flow diagrams. All that is required to navigate to the details of interest is a double click on the phase of interest and then on the activity and step, to navigate to the details of interest. The presentation style is generally an abbreviated form that does not require maintenance of hundreds of pages of text, although one may add such details with the steps. Thus, customization for local preferences is relatively easy, compared to the old methodology bookshelf.

SSI has a special extra bonus, its MAP/Estimator package. This is a Windows-based companion tool to the MAP/Administrator designed for the project manager. After entering a few parameters about the difficulty of the project, the manager can press a "go" button. Lo and behold, out comes a complete program plan in the selected project management package's format, for example Timeline from Symantec Corp. Cupertino, Calif., or Project Workbench from Applied Business Technology, New York City. Imagine the power of being able to generate a detailed workplan automatically that shows why the project cannot be done in six months unless the company's standard implementation approach goes out the window! The previous DOS version of MAP/Estimator was a bit clunky to use, but SSI made good on the promise to convert to the Windows environment.

TOOL IN ACTION

The theory is sound. How does this work in practice? I talked to a couple of SSI users, and got the following reports.

My first contact was Jack White, manager of platforms and processes for IBM's ISSC (outsourcing) business with Southern Pacific in San Francisco. White reported that he selected SSI's methods and methodology assistance tools for that customer. He said ease of use, with the PC tools, and extreme depth of the methodology were strong factors in selecting SSI for a methodology vendor. Southern Pacific's primary interest is reengineering, not new development. So far, it has

Structured Solutions buys rival from EDS

By Dean Anason
Staff Writer

Atlanta-based Structured Solutions Inc., a company which sells software to businesses who want to write their own programs, bought its toughest product line competitor from Dallas-based Electronic Data Systems Corp. on Sept. 18.

The deal included a cash payment to EDS and a royalties arrangement, although exact dollar figures were not disclosed. Structured Solutions anticipates revenues around $5 million this year, which could possibly double next year with this acquisition from EDS, says Robert Connolly Jr., Structured Solutions' president.

The 9-year-old company has now come full circle in its purchase of the "information systems methodology" known as Stradis, Connolly says. McDonnell Douglas Systems Integration Co., which owned Stradis

STRUCTURED SOLUTIONS

before EDS bought McDonnell late last year, was Structured Solutions' first major customer.

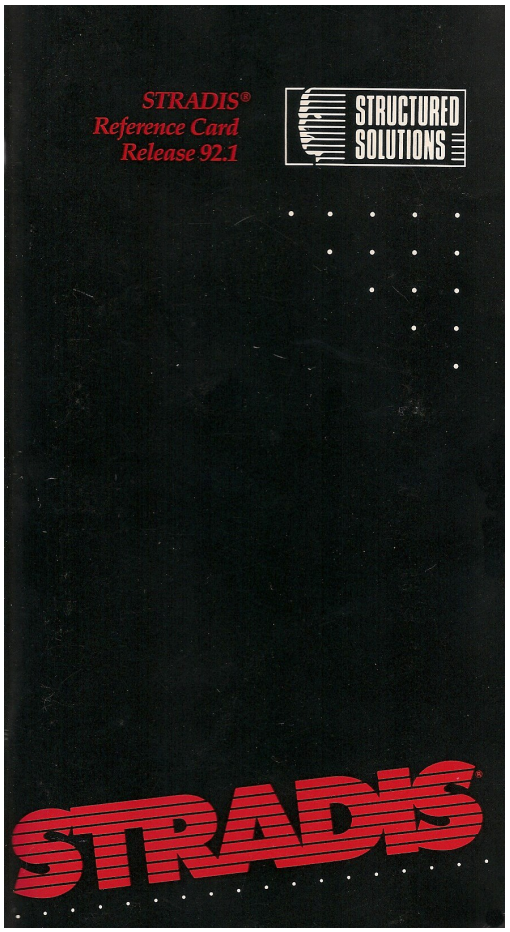
"We basically just bought the company we used to work for," says Connolly, who started Structured Solutions in Tampa, Fla., before moving it to Atlanta two years ago. The company's third owner is Philip Caltabiano, a former vice president of U.S. sales and international sales at Atlanta-based KnowledgeWare Inc.

Although Stradis and Structured Solutions' AD/Method lack the name recognition of a Microsoft Windows, large corporations such as Sprint and Ryder System Inc. use methodology software to design software that performs the most basic functions of their businesses.

A methodology is much like a recipe or instruction booklet for engineering software systems that can come in book form or on computer disk.

Within the software industry, the acquisition will give the Atlanta company a big boost, Connolly says. It includes a customer base of about 140 Fortune 500 companies and a strong presence in international markets such as Britain, Holland and some South American countries, according to the company.

Structured Solutions' principal competition includes the Big Six accounting firms and a number of small companies like itself, Connolly says. He expects the acquisition to help propel the company ahead of the smaller companies, he says.

"We certainly are not an industry household name, but Stradis is," Connolly says. "We felt we needed a shot in the arm to give us some recognition."

Connolly expects the company, which has 19 employees now, to reach 30 by the end of the year, he says. The company plans to maintain the Stradis office in St. Louis and open a new office in Washington, D.C. ■