Winning the Technology Trifecta:

A Software Development Process
that Succeeds at Quality, Timeliness and Cost

Philip Nestingen, CEO
4192 S. Fir Circle
Evergreen, CO 80439
303.941.2495

Winning the Technology Trifecta:
A Software Development Process
that Succeeds at Quality, Timeliness and Cost

EXECUTIVE SUMMARY

To compete in a volatile, global market, executives and business owners must find ways to strategize, optimize and leverage their technology in order to increase efficiency, effectiveness and profitability.   The software infrastructure is the lifeline of any enterprise, large or small, and undergirds all business systems whether sales, accounting, customer service, or marketing. Survival and growth of a company rests on the bedrock of its software. It also can serve as the kinetic energy that catapults them to the top as leaders in their industry. **This requires software that is rapidly produced, within budget, and of the highest quality.**

Until now, achieving all three goals within the software industry has never been fully realized because the other approaches have fallen short of meeting all aspects of the technology trifecta. Dynafactor^{TM} encapsulates the complexities inherent in the software development process and pulls them together into simplistic, yet highly malleable, building blocks. The result is software built with high quality code, in remarkably less time and with controlled cost.   This white paper describes a code generation tool within the software development process called Dynafactor^{TM} - one that allows businesses to achieve the ultimate technology trifecta of quality, timeliness and cost.

INTRODUCTION

According to the Small Business Association, an estimated 552,600 new employer firms opened for business in 2009, and 660,900 firms closed. While numerous reasons can account for failure, there is no question that software infrastructure in a business can serve a critical aspect for business success or failure.

Companies of all shapes, sizes and ages use software for transacting business whether for sales tracking, accounting, payroll, etc. and they currently have three means to satisfying their

software needs. They can purchase COTS software (Commercially Available Off-The-Shelf) which is defined as non-developmental item (NDI) that is both commercial and sold in substantial quantities in the commercial marketplace, and that can be procured or utilized from the general market. The second alternative is to secure the services of a software development firm or employ specially trained IT staff to design, develop and deploy custom, tailor-made software. Finally, businesses may take a quasi-custom approach to develop and employ what is called wizard-generated software. For this method, software tools are used to point to a database and allow the user or business expert to select pre-designed components for building their software in a canned, automated fashion.

From large corporate executives to small business owners, the decision on how best to handle software is critical to the success as a thriving, cutting edge enterprise.   In this era of intense competition, executives must pay attention to the type, quality, and grade of software used to support their business activities. From large corporate executives to small business owners, the decision on how to handle software needs is also undergirded by the fiscal and administrative needs to achieve quality, timeliness and cost. Winning at all three, while sometimes suggested as not being achievable, is crucial for success as a thriving, cutting edge business or corporation.

Achievable? Yes! It's the technology trifecta; quality, timeliness and cost and Nestingen, Inc. has created a development approach that meets the software needs of any organization, large or small.   Dynafactor^{TM} is new technology and is the grail for succeeding at quality, timeliness, and cost in the software development process.

OTHER SOFTWARE DEVELOPMENT ALTERNATIVES

Commercial Off-The-Shelf (COTS) Software

In desperation for a solution, businesses might reach for a software package that is pre-generated and meets the immediate need of their organization.   COTS is helpful to a business and certainly can save time and money in the short run.   Simply buy it off the shelf, install, and deploy the package. However, it is far from meeting the intricate software needs of the enterprise environment and, depending on the solution, may not be the most cost effective approach when calculating a company's ROI.

Furthermore, businesses can solve basic software needs by buying COTS   software and it may work for some fundamental purposes. However, business growth is limited with most COTS software. For example, a business may find their business segmented into application silos and software that only scratches the surface in solving their problems. Furthermore when they look deeply at their systems they find it to be segmented into people silos making its processes and reporting utterly impossible.
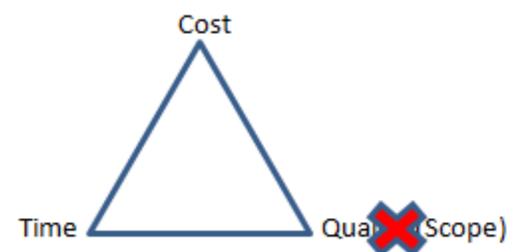
Analogy

COTS software is analogous to a football team that cannot score because no one player is coordinating a common game plan and every player is running their own routes similar to a group of teenage boys playing pickup football. It doesn't take tremendous planning and organizing to play pick up football, but it can't compete with the pros.

Custom Software

Custom software development can certainly help businesses in ways that standard, mainstream systems cannot. Yet, every business is different from employers and employees, to locations and industry specialties and there is no "one-size-fits-all" category when it comes to business solutions. This is where custom software development is valuable.

Custom software development can improve productivity, regardless of the industry. There will be less time tweaking and managing less-than-efficient programs means more time working the business instead. Elimination of application silos and disconnected data brings efficiencies and continuity to a business presence. While some standard solutions DO work effectively, by and large, standard solutions are just that – standard.
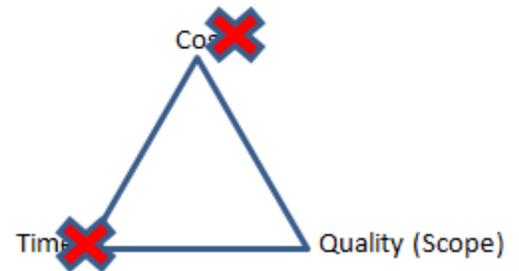
Software development teams are charged with designing and developing user-specific software, but the challenge is usually riddled with problems in meeting any one, or even all three, of the technology trifecta. In "old school" enterprise software development, projects are often plagued with costly problems (no pun intended). Here are just a few problems encountered by traditional custom software approaches.

- Old school software development practices (OSSDP) often run over-budget.

- OSSDP can be inflexible and unpredictable.

- OSSDP is rarely completed on-time or on-schedule.

- OSSDP may not include all the features you need or want.

- OSSDP may no longer serve the business purposes when parameters change.

A business that employs a custom software package approach may find itself caught in the middle of a torrent, deep in a software development project that often fails in the end either by not meeting the needs of the business or causing the company to reach a progressive ceiling because of limited outgrowth potential built within the software. In the end, the time, energy and sometimes inadequate custom solutions, divert important time and energy away from what a company is set out to do in the first place, i.e., fulfilling a business's core mission and goals.

Winning the Technology Trifecta: A Software Development Process
that Succeeds at Quality, Timeliness and Cos
© January 2012 Nestingen, Inc.

Analogy

Custom Software is analogous to a football team that tries to draft the perfect players for each and every position. This would take years to accomplish and by the time they get a great quarterback in place, their receivers are out-of-date and their running backs are worn out.

Wizard-Driven Software

Wizard-driven software provides businesses with tools to generate their "custom" software. There are numerous template designer programs available with the same result as an Access wizard-created interface. However, wizard-driven software does not meet the quality aspect for software and may not even meet timeliness factor. Furthermore, grid- based pop-up application interfaces provide little or no real flexibility in the interface and leaves the user with clunky, ineffective software. Still others will try a wizard driven approach which allows for the specialization and conforming of the software to the general specific needs of the business, only to find it falling short of the real needs of the software – one that fulfills a business's unique model. It might be fast to deploy from concept to in production, however the constraints of their design environments leave a business with little more than a high-bred COTs program.
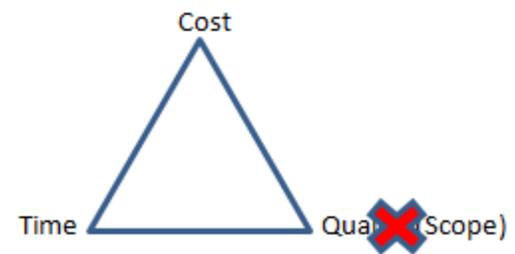
Analogy

Wizard-driven software is analogous to a football team arriving in the red zone only to settle with a field goal. It saves money and time, but falls short of the big score. It simply does not get the job done.

DYNAFACTOR ᵗᵐ - Simplify the Complex

OVERVIEW: Dynafactor™ is a drag-and-drop, declarative, IntelliSense development model. It is an approach whereby the complexities of the environment are encapsulated and pulled together into a set of simplistic, yet highly malleable, widgets. It can be described as a set of building blocks similar to Lego© blocks at a granular level with blocks that are small enough to allow for very robust design and development.

It is….

Robust

Declarative

Malleable

Comprehensive

Intuitive

Dynafactor™ works through 14 steps to complete its task:

Step 1. Dynafactor™ validates the data model and verifies that a database includes primary keys on all tables.

Step 2: It builds all of the Stored Procedures and Triggers necessary to handle all of the CRUD

Step 3. It builds a table and row ORM for each database table which includes the CRUD for each foreign key.

Step 4. It creates static variables for each column for every database table.

Step 5. It prepares and completes a project file and corresponding file structure for storing all elements of the ORM.

Step 6. It qualifies the set of files against "Team Foundation Server."

Step 7. It builds a succinct, detailed widget for each table that includes validation and state management using a proprietary layout management system.

Step 8. It generates a grid-widget based on user-specified rules.

Step 9. It places all widgets into a separate project.

Step 10. It prepares and completes a project file and corresponding file structure for storing all elements of the widget project.

Step11. It qualifies the set of files against "Team Foundation Server."

Step 12. It takes the separate ORM and widget projects and includes them within a solution file.

Step 13. It qualifies the set of files against "Team Foundation Server."

Step 14. It is ready for a developer to complete the final project using Visual Studio.

WINNING THE TECHNOLOGY TRIFECTA:

Why is Dynafactor^{TM} so important and useful in the software development arena? Simply put, it wins the technology trifecta.

QUALITY

ERROR-FREE: NI's advanced technology is built specifically to reduce the number of bugs and almost simultaneously find and fix them.   Codified patterns built within our technology supports a virtually flawless development environment.   A software project previously deployed with NI solutions have operated flawlessly for life without error.

INTRODUCES PREDICTABILITY: Dynafactor's program design is stored as meta-data thus allowing it to be declarative, repeatable, and consistent with the long-term goals of the company.

ALLOWS CONFIGURABILITY: Deep, surface, or cosmetic changes within the database structure and application are easily supported and facilitated with Dynafactor's organic modeling approach.

CONSISTENT, PROFESSIONAL LOOKING INTERFACE: Fabricated applications have highly consistent and professionally designed user interfaces, giving your applications a finished look and feel even though they may be prototypes or works-in progress. Users can better visualize the finished application and will confer a degree of perceived quality typically not present in traditional hand-coding development until the end of the project.

STABLE CODE: High-quality generated code is very stable, reducing a QA burden. Businesses can take advantage of the testing already applied to the application generation tool.

LEGACY-FREE: Dynafactor^{TM} uses a "legacy-free" development process allows companies a chance to maximize their strength and filter out dated processes.

ENGAGES & EMPOWERS COMPANY USERS: Dynafactor's "intuitive preference architecture" allows users the ability to easily control their experience and ultimately puts the company in control of the software outcome.
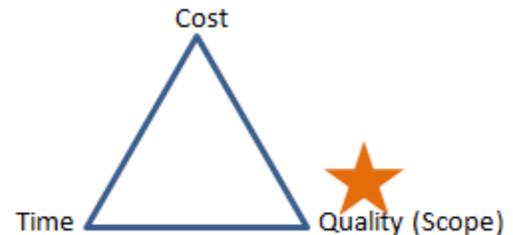
TIMELINESS:

CONCEPTION: The sophisticated "widget and canvas technology" allows for extremely fast custom software development from conception to deployment. It generates fully workable application building blocks that can be used to quickly mock-up the application using a database schema as input, skipping much or all
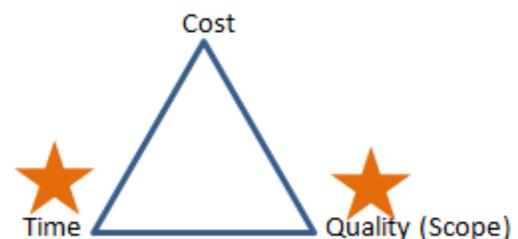
Winning the Technology Trifecta:

that Succeeds at Quality, Timeliness and Cost

© January 2012   Nestingen, Inc.

of the mock-up phase of the project. Using application generation, a business can quickly create prototypes and gather user feedback. This allows organizations to be very responsive to internal customers or impress external clients with one's development capabilities.

WORKFLOW DESIGN: Custom workflow processes designed to streamline a business can be quickly designed, built and deployed. Workflow design is often an iterative process, taking several cycles to ascertain the business process and then to reorganize and streamline it.

SCREENS: Application fabrication lets you quickly refine screens and workflows, engaging application users in a more iterative development process.

PRIORITIZING TASKS: Eliminating large amounts of hand coding lets one to focus on other high-priority tasks and projects. While every organization has different priorities, Dynafactor reduces the time for hand-coding.

MAINTAINABILITY: Dynafactor^{TM}- generated applications follow a highly consistent architecture, allowing any developer to easily maintain any application.

RAMP-UP TIME: There is little or no 'ramp up' time necessary for one developer to maintain another developer's application because the architectural knowledge transfers from one application to another.

IT DEPARTMENT PRODUCTIVITY: Most developers dread building features such as logging security, paginated reports, hierarchical navigation, filtering, and full text search because the amount of time required is disproportional to the value they create. Application users expect them, but they usually require considerable coding validation and layout. An application window may contain 25 or 40 separate SQL queries, each of which has to be written, debugged, and vetted for performance. However, Dynafactor^{TM} allows IT staff to no longer worry about manually programming pages, controls, or SQL. They generate most, if not all, of this repetitive code.

COST:

The old adage, time is money, is never more salient than in the software development process. Nestingen Inc. customers have saved up to 60% per application. How about speed of deployment? What if the project can be delivered in half the time?

It only takes Dynafactor 30 minute to fabricate the code needed for 222 tables of a given database with almost 4,000,000 lines of code. While that is a tremendous amount of code, the architecture of the code is such that performance is not hindered in the least.   Let's say a developer writes 100 lines of code per day.   At that rate, it would take about 174 developers one full year to duplicate what Dynafactor can do in 30 minutes.

Name Space	Object	Lines
Total	9,665	2,097,360
ORM	1,288	1,807,064
Alive	609	15,528
Store Procedures	7,768	3,919,952

Analogy

Dynafactored software is analogous to an elite football teams with expert coaching, unbelievable quarterback and a tremendous defense line. All together, they make football history and a winning team!

CONCLUSION:

Winning, whether in football or the market, requires ingenuity, talent, responsiveness and true grit. It is everything to anyone who wants to win the business game and when winning is everything, companies can't afford to fall short in any one element of the technology trifecta. Dynafactor^{TM} is a game changer in today's competitive playing field. Not only does it win the technology trifecta, but so do businesses, large or small.

What Dynafactor^{TM} can do for a software project is unprecedented.   Dynafactor^{TM} was built with the needs of a winning business in-mind. It addresses the age-old problem of managing cost, bouldering features, and responding in flash speed. Winning teams are always on the look-out for pushing the envelope on what it *can* do, not what it can't do. Dynafactor^{TM} can do what every business needs it to do…help them to win the technology trifecta and ultimately win at efficiency, effectiveness and profitability - the ultimate enterprise trifecta.