



Research, Technology, and System Solutions

Developing Architecture-Centric Engineering Within TSP

Bursatec, the technology arm of Grupo Bolsa Mexicana de Valores (BMV, the Mexican Stock Exchange), recently completed a project that replaced three existing trading engines with one system developed in-house. Given the competitiveness of global financial markets and recent interest in Latin American economies, Bursatec needed a reliable, fast new system that could work ceaselessly throughout the day and handle sharp fluctuations in trading volume. To meet these demands, the SEI suggested combining elements of its Architecture Centric Engineering (ACE) method, which requires effective use of software architecture to guide system development, with its Team Software Process (TSP), which teaches software developers the skills they need to make and track plans and produce high-quality products. This article describes the challenges Bursatec faced and outlines how working with the SEI and combining ACE with TSP helped it address those challenges.

Challenges

The team of Bursatec software architects faced a significant challenge in designing the new trading system: only one team member had significant experience in designing a financial software system. We felt the ACE methods would help the team better understand what software architecture means, particularly for thinking about abstractions and solving quality attribute problems. Another complicating factor was that Bursatec

wanted to combine stock market trading with derivative market trading on the same platform, to reduce operating costs and provide a single interface to external financial markets that afforded high throughput, low latency, and high confidence.

Getting Started

One of our first steps was to conduct a Quality Attribute Workshop in which the Bursatec stakeholders defined the five most important quality attribute requirements (also known as quality attribute scenarios) their new trading system had to fulfill. To guide the system design, the Bursatec architecture team used the Attribute Driven Design (ADD) method. ADD is a decomposition method based on transforming quality attribute scenarios into an appropriate design.

Not surprisingly, given the importance of speed for the new system, the stakeholders identified runtime performance as one of the most important quality attribute scenarios. The performance quality attribute scenario, coupled with high availability requirements, led the team to realize that conventional approaches, such as a three-tier architecture, were not the best solution for its new system. Consequently, the architecture team spent the next two weeks exploring various solutions, as well as the potential negative outcomes of each proposed solution.

At the end of the two weeks, using rigorous Architecture Tradeoff Analysis Method (ATAM) techniques, the team of Bursatec architects was required to present its findings—as well as evidence (including measures) that its chosen approach was correct—to an SEI software architecture coaching team that challenged each scenario. Every subsequent two weeks, the Bursatec software architects presented solutions with appropriate evidence for the scenarios they had created. For example, with respect to the performance requirement, the team demonstrated how a stock order would traverse the system, estimating and measuring the timing required for every step. With each review, SEI coaches identified risks associated with a particular approach. For example, the team identified one risk with respect to performance: synchronizing with backup systems would throw off the timing. In all, there were three iterations of the architecture, each lasting six weeks.

At the start of the second iteration, the SEI software architecture coaches brought in the team of Bursatec developers to begin working on prototypes, specifically focusing on risks (such as the timing of querying complex data structures) that could not be addressed solely via software architecture. This important step allowed developers to deepen their understanding of the architecture and familiarize themselves with the problem, which was a

Research, Technology, and System Solutions

Developing Architecture-Centric Engineering Within TSP

lengthy process. The developers had six weeks to implement the prototypes; at the beginning of the third iteration the developers returned and presented their results to the architecture team. This process enabled the architects to finalize their architecture design using the results from the prototypes.

An interesting benefit to this style of architectural coaching was that the Bursatec architects used Enterprise Architect (a Unified Modeling Language-based tool) from the onset, to document, evaluate, and justify their solutions. Although architecture documentation is often an afterthought, it became second nature to the Bursatec architects. The architects focused only on the documentation that was either (1) needed to provide sufficient evidence that the system would support the quality attribute requirements, or (2) was required by the developers to effectively implement prototypes and the subsequent system.

Improving Delivery with TSP

The Team Software Process (TSP) is a team-centric approach to developing software that enables organizations to better plan and measure their work and to improve software development productivity to gain greater confidence in quality and cost estimates. Our coaches emphasized the incorporation of TSP principles throughout the architecture design process with Bursatec. The use of TSP enabled the Bursatec architects to prepare, estimate, and track their work. In this case, the Bursatec architects were also able to time-box their iterations, an approach that the SEI finds effective. These activities initially proved challenging because TSP is oriented toward programming, so the measures employed by

developers typically apply to lines of code, classes, requirements pages, or other tangible, implementation-oriented measures. To create a measurable work unit, the Bursatec architects used the quality attribute scenarios as a size measure.

The SEI architecture team recognized that each quality attribute scenario would be refined into about five detailed quality attribute scenarios that address special cases. The SEI team also recognized that the Bursatec architects would have to create at least three to five diagrams and descriptions to fulfill each scenario. The Bursatec team then estimated how long it would take to create each diagram with a description. We found this measure proved a good tool for determining the time required to complete the architecture, which had been difficult to estimate in our prior work with organizations. This approach to measuring and estimating work allowed the Bursatec architects to provide accurate estimates of deadlines to their management team.

Integrating the ACE architecture within the TSP management process gave the Bursatec architects an effective framework in which to work. While it did restrict some of their freedom, it also proved helpful. For example, the architects' work was structured into iterations, each with different goals. The first iteration focused solely on discovering problematic areas of the system based on achievement of the necessary quality attribute scenarios. In subsequent iterations the architects gradually added details to the system design to include support of all quality attribute scenarios. This iterative method enabled them to create a software architecture organically, for the whole

system, that was well understood, justified, and accepted by the team.

Building and Evaluating the System

Once the architecture was complete, the SEI architecture coaching team conducted an active design review in which the Bursatec architects communicated the entire architecture to the developers in a structured way. Next, conformance reviews were conducted during which the developers needed to provide the architects with evidence that the systems they were building conformed to the architecture. These reviews reinforced confidence that the whole system would meet the needs of the stakeholders.

The development of the new trading system for Bursatec progressed on schedule and within budget. Moreover, tests confirmed that the trading system performance far exceeded expectations. The combination of TSP and ACE proved an ideal approach for the development of the trading system. TSP brought discipline and measurement, while ACE provided a set of robust architectural techniques that focus on business goals and quality requirements. Both approaches together supported the whole development lifecycle, emphasizing business and quality goals, engineering excellence, defined processes, process discipline, and teamwork.

By Felix Bachmann, Senior Member of the Technical Staff

Related Web Sites

www.sei.cmu.edu/tsp
www.sei.cmu.edu/library/abstracts/reports/10tr031.cfm

For Course Registration

www.sei.cmu.edu/products/courses/

Research, Technology, and System Solutions

Developing Architecture-Centric Engineering Within TSP

For General Information

For information about the SEI and its products and services, contact

Customer Relations

Phone: 412-268-5800

FAX: 412-268-6257

info@sei.cmu.edu

www.sei.cmu.edu