

## Agile Approach

Improves organizations through agile methods. Tackles the difficult problems of sustainable software development and software scalability. Breaks down barriers that divide business, development, QA, and operations.

Approaches all software as a shippable product that provides essential value to its users. Employs a simplicity strategy to achieve this goal. Enterprise software projects can succeed by adopting a product perspective.

### Agile Engagement - Most Recent

- **Solution Areas:** Applications, Integrations, Scalability, Performance, DevOps, Methodology, Release Management
- **Roles:** Application Architect, Integration Architect, Systems Architect, Performance Architect, Problem Solving Lead, Agile Developer, Agile Facilitator, Software Engineer, Automation Engineer, Technical Lead, Middleware Mentor, QA Mentor, Build Engineer, Release Engineer, Deployment Engineer
- **Methods:** Simplicity, Innovation, SCRUM, Sprints, User Stories, Iterative, Incremental, Personal Kanban, Agile Manifesto, Extreme Programming, Emergent Design, Evolutionary Architecture, Deployment Empathy, DevOps

### Agile Development - Most Recent

- **Architectures:** JEE, J2EE, MVC, MVP, Hexagonal Architecture, Web Services, Integration, Messaging, Caching
- **Techniques:** Architectural Spikes, BDD, DDD, TDD, Design Patterns, Refactoring, AOP, Fluent API, Immutability, Thread-Safe Code, Mock Objects, Test Doubles, Test Harness, Continuous Integration, Unit Testing, Integration Testing, Performance Testing, DRY, KISS
- **Tools:** Java, SQL, C, Bash, Eclipse, Bamboo, Confluence, JIRA, CVS, SVN, Linux, Windows, Maven, Ant
- **Frameworks:** Spring DI, Spring AOP, Spring Web MVC, Spring Web Flow, Spring Integration, Apache Struts, Hibernate, DBUnit, JUnit, JMock, many more
- **Servers:** Apache Tomcat, Apache HTTP Server, Oracle Database, Oracle OAS, Oracle WebLogic, Oracle OHS, Oracle WebCache

### Technology Evaluation and Adoption Research

- **Architecture:** LMAX, CQRS, Event Sourcing, Mechanical Sympathy, Polyglot Programming, Polyglot Persistence, Micro-Services, Single Page Web Applications, Javascript Micro-Frameworks, Logic-Free Markup
- **Techniques:** Functional Programming, Storage Services, Services not Layers, Eventual Consistency, Continuous Deployment, Production Immune System, Infrastructure as Code, User Journeys
- **Tools / Frameworks / Servers:** Clojure, Memory Grids, Git; JQuery, JQuery UI, JQuery Mobile; node.js, Nginx, Webbit; many more
- **NoSQL Tools:** (Document) CouchDB, MongoDB; (Graph) NeoJ, InfiniteGraph; (Key-Value) Cassandra, Riak, DynamoDB, Coherence, Hazelcast, Redis, BigTable, MemcacheDB, MongoDB, Hadoop, Datomic

### Simplicity Strategy

- **Invests in simplicity and sustainability** as vital for **emergent quality** and **continuous improvement** of ROI
- **Moves troubled projects** to improved architectures, technologies, and practices, **minimizing risk with Agile methods**
- **Empowers critical thinking** by **skilled workers** through **automation of repeatable tasks and processes**
- **Adopts a polyglot technology strategy** to create solution patterns tailored to **organizational context**
- **Invests in knowledge capture** through **documentation as code** and **periodic retrospectives**, not project copying
- **Simplicity begins slowly** as contexts are defined; **delivery speeds up** over time due to **emergent flexibility** of **solid design**; an **easy approach degrades** in efficiency over time due to **emergent complexity** from **ad hoc choices**

## Career Highlights - Enterprise Architecture

### CONSULTING SOFTWARE ARCHITECT / Health Care Services

*Fresenius Medical Care / Lexington, MA*

**November 2005 - Present**

Currently providing consulting services in support of world-class clinical services offerings, including Chairside, the **largest known deployment of a clinical services application in the world today**. Intensive focus on **hands-on solutions development** for applications, integrations, scalability, performance, DevOps, methodology, and release management, while simultaneously serving as **Agile Architect** supporting the aggressive adoption of **sustainable software development**. **Technical Lead** across many years of application releases.

### INFRASTRUCTURE TEAM LEADER / Student Information Systems IT

*Massachusetts Institute of Technology / Cambridge, MA*

**July 2001 - November 2004**

The student information system at M.I.T faced a crisis: The client-server application platform lacked scalability and maintainability. As architect and team leader, drove departmental strategic planning, migration to a **J2EE** enterprise computing strategy, adoption of agile project methodologies and an IT services framework. Designed, planned, procured, installed, configured, deployed, tested, and managed a modern and flexible SOA platform (**Apache**, **WebLogic Server**, **WebLogic Portal**, Hyperion IS, Hyperion SQR, **Oracle Database Solaris**, **Linux**).

### ENGINEERING ARCHITECT / Enterprise Intelligence Applications

*Painted Word / Cambridge, MA*

**November 1999 - December 2000**

Painted Word needed a cost-effective web applications strategy to enhance enterprise intelligence offerings to Fortune 1000 clients. As the lead **J2EE Architect** during a strategic transition from products to services, developed the Painted Word **Java Framework** for **MVC web application** development that served as the cornerstone for a highly successful thin-client web applications strategy (**UNIX** and **Windows** platforms) and **several million dollars in core revenue**.

## Career Highlights - Technical Architecture

### CONSULTING SOFTWARE ENGINEER / Telecommunications Messaging Service

*Comverse Network Systems / Wakefield, MA*

**October 1996 - July 1999**

Comverse provided messaging systems to telecommunications companies worldwide. As architect and team leader, delivered a common codebase with hard real-time messaging and on-demand account migration for 24 x 7 distributed UNIX platforms, essential to **primary revenue contracts** (e.g. **\$70 million**).

### CONSULTING SOFTWARE ENGINEER / Financial Instruments Database Services

*Fidelity Investments, Vantage 20/20 / Boston, MA*

**September 1995 - September 1996**

The Vantage 20/20 project to synchronize legacy brokerage systems to a UNIX / Oracle platform had been blocked by data integration issues for a year. As analyst and developer, resolved core issues through the redesign of the broker data model, data house-holding processes, and daily round-trip database synchronization. Set new standards for virtually 100% code coverage, while piloting Oracle Designer.

## Enterprise Architecture Experience

### CONSULTING SOFTWARE ARCHITECT / Health Care Services

*Fresenius Medical Care / Lexington, MA*

November 2005 - Present

Currently providing consulting services in support of world-class clinical services offerings, including Chairside, the **largest known deployment of a clinical services application in the world today**. Intensive focus on **hands-on solutions development** for applications, integrations, scalability, performance, DevOps, methodology, and release management, while simultaneously serving as an **Agile Architect** supporting the aggressive adoption of **sustainable software development**. **Technical Lead** across many years of application releases.

#### Software Architecture:

- **Executed mission-critical platform migrations** of the Chairside application, which included Oracle Application Server (OAS) 10.1.2 (J2EE 1.3) to OAS 10.1.3 (J2EE 1.4), and OAS 10.1.3 (J2EE 1.4) (HP-UX) / Oracle Database 10g (HP-UX) to Oracle WebLogic Server 10.3.5 (JEE 5) (Linux / VMware) / Oracle Database 11g (AIX / LPAR).
- **Executed architectural spikes** that included platform migration prototypes from Oracle Application Server (OAS) 10.1.2 (J2EE 1.3) to OAS 10.1.3 and JBoss 4 (J2EE 1.4), security messaging solutions, and distributed caching mechanisms.
- **Instrumental in architectural bootstrapping, architectural spikes, and final implementation** of the current Chairside application architecture, design patterns, and implementation idioms, which leverages leading frameworks such as **Spring AOP** for dependency injection, **Spring Web Flow** for declarative workflow, **DWR** for AJAX support, **Spring ORM** support for **Hibernate**, and **Spring JDBC** support, amongst many others.
- Served as **technical leader for architectural refactoring** of the Chairside application web layer from implementing **Model-View-Controller (MVC)** architecture (**Struts**) in the legacy release to **Model-View-Presenter (MVP)** architecture (**Spring Web Flow**) in the current release. This **simplified the coding** and provided for **improved separation of responsibilities**.
- Served as **technical leader of set-based design efforts** for architectural spikes to solve client-visible issues in the Chairside Dialysis Machine Observer applet through brainstorming of solutions for **incremental refactoring** and wholesale **architectural replacement**.
- **Instrumental in architectural bootstrapping** of the Chairside Machine Data project to provide a centralized polling solution to replace the Dialysis Machine Observer browser applet for collecting treatment data from dialysis machines in the field and making it accessible to the Chairside application through integrated AJAX web service calls in the browser.
- Supported **integration architecture** solutions between the Chairside application and the corresponding system of record for both the legacy release and the current release.
- Collaborated on **integration architecture** with the shared services team to incorporate and optimize **messaging solutions** and **web services** in support of security across multiple clinical applications.
- **Evolved an Architecture Owner role, as an Agile Architect**: facilitating creation of the architecture, not enforcing it, transitioning architectural skills to other team members, executing architectural spikes, mentoring team members in organizational technical guidance, breaking decision deadlocks, and reviewing architectural work from other teams.
- Served as the primary resource for **final architectural and design guidance** for the clinical applications team.

**Scalability and Performance:**

- Served as **performance architect and technical leader** for **scalability and performance tuning** of the next generation Chairside application platform, a **Two-Tier Proxy Architecture** consisting of a Proxy Tier composed of stacks of Oracle Web Cache Servers (Linux / VMware) and Oracle HTTP Servers (Linux / VMware) and an Application Tier composed of an Oracle WebLogic Server Cluster 10.3.5 (JEE 5) (Linux / VMware) and an Oracle Database Server Cluster 11g (AIX / LPAR).
- **Designed test strategies and test scenarios.** Coordinated testing and data collection across multiple teams for all tiers. Led analysis of test results and recommendations for tuning changes. Leveraged **Coradiant, OpNet, Oracle Grid Control, WebLogic Console**, and other tools for test metric data collection. **HP LoadRunner** was used for load testing.
- Served as **performance architect and technical leader** for **failover testing** of the next generation Chairside application platform. Created **load balancer monitoring and failover strategy** applicable to a Two-Tier Proxy Architecture. Created user stories with robust expectations and success criteria for all tests.
- Served as **performance architect and technical leader** to tune the legacy Chairside application platform core workflow. Recommended **separation of services** onto distinct nodes for **tuning isolation** according to resource capacity needs. Proposed, formulated, and led a **wait-based tuning initiative** to reduce the JDBC pool connections by an order of magnitude to achieve optimal database performance. **Minimized database load while enabling high throughput** with insulation from sporadic high latency database latency spikes. **Improved scaling of the application architecture to support deployment from hundreds to thousands of clinics.**
- Served as **performance architect and technical leader** to tune the legacy Chairside application platform printing services. **Improved performance of JVM garbage collection** to minimize stop-the-world events and prevent entry into a state of continuous 100% CPU utilization.
- Served as performance architect for **tuning web services** for security owned by shared services team. Leveraged **Coradiant** to monitor application and service response times, which identified security web services as a potential root cause of Chairside latency. Recommend an increase in **Oracle HTTP Server (Apache) connection pools** that **eliminated daily multi-second latency spikes and workflow interruptions** in the Chairside application.
- Proposed architectures for **shards of dedicated capacity** for security services to isolate Chairside from other clients.
- **Solved numerous customer-visible concurrency issues in legacy production code.** Solutions included correcting threading issues in a browser applet that interfaced with medical device instrumentation, correcting use of non-thread-safe Java SDK objects, and correcting use of non-thread-safe XML parser objects.
- **Discovered platform defects as the root cause of recurrent performance degradation** that emerged under accumulated message load in the Chairside distributed caching mechanism, which leveraged Oracle Advanced Queuing. Proposed alternative caching architectures to circumvent the problem.
- Identified the instantiation of **multiple Spring Application Contexts** as the **root cause of subtle concurrency defects** in several application configurations.
- Served as the primary resource for **scalability and performance tuning guidance**, as well as **critical problem solving leadership and insights**, for the clinical applications team.

**Sustainable Software Development:**

- **Supported the adoption of SCRUM** methodology for agile project management, as both **Agile Developer** and **Agile Facilitator**. Participated in tooling proof of concept efforts. Incorporated the Atlassian software stack into the development cycle, leveraging **Bamboo** for continuous integration, **JIRA** for user story and task management, and **Confluence** for written collaboration and knowledge capture.
- Assisted in mentoring other team members in the use of **SCRUM** artifacts and processes, including **task boards, product backlogs, sprint planning, sprint backlogs, user stories, task estimation, daily huddles, sprint reviews, sprint retrospectives**, and **periodic backlog grooming** with the product owner. Tempered the use of processes and tools through the explicit adoption of the **Agile Manifesto**.

Strategic Leader • Critical Thinker • Hands-On Coder • Agile Architect • Continuous Idea Generator

- **Incorporated architectural spikes** into the development cycle to reduce project risk and enable the breakdown and estimation of epic user stories as needed.
- Incorporated **extreme programming practices** into the development cycle, experimenting as appropriate with varying levels of **pair programming**, **test driven development**, **continuous integration**, **refactoring**, and **simple design**, in order to improve communication, simplicity, feedback, courage, and respect. Achieved the highest levels of productivity ever seen on the Chairside project during sustained **SCRUM** execution.
- Adopted the use of **mock objects** to support **unit testing**, and **test doubles** for **integration testing**. Created **test harnesses** that leveraged the **Spring DI** framework for out of container **integration testing**. Created a **test data generator** tool to support **test doubles** for repeatable **performance testing**.
- Advocated the adoption of **domain driven design** and **fluent API** development principles to support the robust encapsulation of object responsibilities. Leveraged **AOP** to enable a **hexagonal architecture** style of development. Adopted **immutability** principles to ensure **thread-safe code** instead of synchronization, wherever possible.
- Advocated the adoption of **performance testing** as a mandatory **regression test** for each release and **continuous performance testing**.
- Adopted the **KISS** (Keep It Simple, Stupid) strategy to fully embrace **emergent design** and **evolutionary architecture**.
- Prototyped leveraging **VMware** player to run **virtual client images** to execute **selenium test cases** for development.
- **Automated generation of LoadRunner scripts** per specific test environment to enable simulated production workflows, after researching which artifacts should be version controlled.

**Technical Leadership and Mentoring:** Technical Lead, Middleware Mentor, QA Mentor

- Served as **technical lead for dozens of releases** of all versions of the Chairside application, actively developing **key features for each release**, with development work spanning all application layers.
- **Mentored QA in best practices** for defect reporting. Assisted QA to model real-world user workflow for load testing.
- **Supported QA with tooling** for a **test data generator** to enable **repeatable load tests** against **system of record test doubles** for integration sources.
- **Assisted QA with platform capacity and network requirements** to enable repeatable load tests.
- **Mentored Middleware team in best practices** for systems administration, performance monitoring, and performance tuning.
- **Assisted Middleware team in capacity planning** exercises for Chairside and other applications.
- **Created Middleware deployment documentation** for the Chairside application.

**Release Engineering:** Build Engineer, Release Engineer, Deployment Engineer,

- Assisted in the **proof of concept** and selection of **continuous integration** tools. Created numerous **build plans** for both **Pulse** and **Bamboo**.
- **Created and refactored build scripts** leveraging either **Ant** and **Ivy** or **Maven** for numerous applications.
- Refactored applications to support **deployment portability** across many environments.
- Created **build process automation** and **deployment process automation**.
- Separated **build process automation** from **deployment process automation** by leveraging **living artifacts** under **version control**.
- Designed a **release management strategy** to support **parallel feature development** with **just-in-time feature freezing** that is also possible to be fully automated. This strategy **limits the possibility of file conflicts during automated merging** by assisting developers to **reset feature branches to the latest production branch**.

**INFRASTRUCTURE TEAM LEADER / Student Information Systems IT***Massachusetts Institute of Technology / Cambridge, MA***July 2001 - November 2004**

The student information system at M.I.T faced a crisis: The client-server application platform lacked scalability and maintainability. As architect and team leader, drove departmental strategic planning, migration to a **J2EE** enterprise computing strategy, adoption of agile project methodologies and an IT services framework. Designed, planned, procured, installed, configured, deployed, tested, and managed a modern and flexible SOA platform (**Apache, WebLogic Server, WebLogic Portal**, Hyperion IS, Hyperion SQR, **Oracle Database Solaris, Linux**).

**Enterprise Strategies:**

- Developed **IT strategic vision** for SSIT and drove key departmental objectives including improved client engagement, business-aligned development, validated and reliable applications, scalable and maintainable application architectures, and robust infrastructure operations.
- Collaborated on **client engagement strategy** to drive entry into risk-driven **project methodology**, improving the accuracy of project definition, sizing estimates, and yearly resource budget planning. Improved client relationships and project predictability through shared business and technical perspectives, agreed project roles, risk assessment, use cases, transparent business-aligned development efforts, and time-boxed delivery.
- Injected **IT architectural governance strategy** through well-defined project lifecycle engagement points for infrastructure team guidance, facilitation, contribution, impact assessments, and alignment reviews.
- Defined **enterprise documentation standards and practices** for project engagement and definition, use cases, test cases, technical documentation, support requests, and maintenance agreements. Mentored project teams; created, reviewed, and audited documentation artifacts, as needed.
- Brainstormed **development team evolution strategy** incorporating expert mentoring by external project leads. Deployed best practices, such as team programming, test-driven development (**JUnit**), repeatable builds (**Ant, CVS**), standardized environments, automated integration (**CruiseControl**), team testing, and continuous customer collaboration. Defined project skill sets in line with methodology, drove J2EE training.
- Designed **IT support strategy** that clearly defined workflow between external clients, application teams, and infrastructure team. Improved prioritization and turnaround of requests. Addressed root causes of production instability by progressively injecting fitness checks earlier into the project lifecycle over time.
- Created and drove **infrastructure migration strategy** to begin transformation of SSIT from a legacy client-server workshop to a modern business-aligned operation built upon enterprise architecture best practices.

**Infrastructure Projects and Operations:**

- Designed **infrastructure migration strategy** to move the MIT Student Information System from a legacy client-server platform (SQL\*Forms, Pro\*C and SQR Batch Jobs, OpenVMS, Netscape Server, CGI, Perl, SQR, Tru64, Stored Procedures, Oracle 7, Solaris) to a modern enterprise application services platform (**Apache, WebLogic 8.1 Server / Portal, Linux, Oracle 9.1, Solaris**) with best of breed components (**JAAS, Struts, EJB Stateless Session Beans, POJO** business logic, **DAO, Hibernate, Quartz**, WebLogic Workshop) integrated with MIT infrastructure (**Kerberos, X.509, Roles**).
- Created identical SOA **production and integration tiers** featuring **failover, load balancing, and SSL acceleration** via mirrored **F5 Big-IP** and **CISCO** appliances fronting a dual-NIC network composed of **WebLogic 8.1 Server and Portal** farms across 6 IBM 335 2 CPU (RHEL 2.1 Linux), **Oracle 9.1** on SunFire v1280 4 (of 12) CPU, **Hyperion IS 3** on SunFire v880 2 (of 8) CPU, and 1.2 TB mirrored storage.
- Achieved 24 x 7 **mission-critical production platform** designed for a 4-year lifespan with standby **Oracle 9.1** instances on alternate same tier machines and physically distant integration tier machines. All systems selectively recoverable from nightly Tivoli online backups against **SAN** storage with offsite tape archives.

Strategic Leader • Critical Thinker • Hands-On Coder • Agile Architect • Continuous Idea Generator

- Built **application platform foundation** to support migration from client-server product lines to enterprise architectures and services. Collaborated with application teams to spin off standardized **J2EE** architecture, application development kit, and development tier environment featuring common scripts for automated builds (**Ant**), deployment (**WebLogic**), change management (**CVS**), and integration promotion.
- **Managed day-to-day operations involving 50 servers** supporting legacy, enterprise, and infrastructure applications. Services included account management, connectivity, printing, data feeds, change management, software configuration, data restoration, systems monitoring, and troubleshooting across all tiers and user communities (students, staff, developers). Collaborated with IS&T on OS, network, and hardware issues.

## ENGINEERING ARCHITECT / Enterprise Intelligence Applications

*Painted Word / Cambridge, MA*

**November 1999 - December 2000**

Painted Word needed a cost-effective web applications strategy to enhance enterprise intelligence offerings to Fortune 1000 clients. As the lead **J2EE Architect** during a strategic transition from products to services, developed the Painted Word **Java Framework** for **MVC web application** development that served as the cornerstone for a highly successful **thin-client web application** strategy (**UNIX** and **Windows** platforms) and **several million dollars in core revenue**.

### Enterprise Strategies:

- Championed **use case driven design** to manage user expectations, requirements and development schedules, resulting in a **lightweight methodology** initiative based on the Unified Software Process.
- Introduced senior consultants to the concept of a **scalable application development methodology**, leveraging the Painted Word **Java Framework**.

### Engineering Projects:

- Developed the Painted Word **Java Framework** to enable **rapid development** of enterprise intelligence **web applications**. This framework directly supported a scalable development methodology.
- Served as project manager and architect for update of MochaBlend **Java API** middleware server (RMI) to support Hyperion Essbase 6.0 OLAP database API.
- Designed strategic plan for **component development practices** to support future projects at Painted Word.

### Revenue Generation:

- The ability to develop thin-client web applications, utilizing the Painted Word **Java Framework**, was **crucial to the business development** of the majority of consulting engagements for year 2000.
- Sale of the Painted Word **Java Framework** source code resulted in a **monthly revenue all-time high**.
- Completed **proof-of-concept thin-client web application framework** for enterprise intelligence applications with web developer under extreme deadline. Demonstrated to the Hyperion CFO, resulting in consulting contract and numerous business development opportunities, including Hyperion application manager project.

## Technical Architecture Experience

### CONSULTING SOFTWARE ENGINEER / Telecommunications Messaging Service

*Comverse Network Systems / Wakefield, MA*

October 1996 - July 1999

Comverse provided messaging systems to telecommunications companies worldwide. As architect and team leader, delivered a common codebase with hard real-time messaging and on-demand account migration for 24 x 7 distributed UNIX platforms, essential to **primary revenue contracts** (e.g. **\$70 million**).

#### Japan Wireless Release Digital Messaging Network Project

**Technical Lead, Platform Group:** This effort required a port of the Digital Messaging Network (DMN) product from the Americas codebase to the Japan Wireless Release (JWR) project. All coding done using gnu C and C++ for SCO UNIX and Univel targets under SunOS ClearCase.

The DMN component implemented a distributed architecture to support real-time messaging services between subscribers on networked voice services systems. Each system was composed of a network of up to dozens of UNIX machines, intelligent switches, and several Windows NT servers. The DMN component was implemented using distributed database and application server architectures.

- Led the platform group DMN team through all phases of the development cycle from requirements to deployment. Coordinated DMN issue resolution between all groups in the release. Mentored programmers unfamiliar with the DMN architecture during all phases.
- Analyzed existing product limitations, and designed solutions for the system requirement requiring 24 x 7 hard real-time operation. Unlike previous projects, JWR required real-time lookup of remote subscriber information prior to message authorization.
- Completion of this effort enabled DMN in the JWR releases for the first time, and satisfied an absolute cornerstone requirement for JWR, **enabling the largest company contracts of the fiscal year**. Only team to finish on time within the release.

**Technical Lead, Workflow Design:** This effort involved the design of a workflow to integrate DMN and Remote Subscriber Migration (RSM - see earlier assignment) for JWR 2.

- Coordinated efforts of all development groups in the release for this design.
- Identified strategy to integrate process workflow in both products with minimal impact on existing code. The impact of all code changes for each group had to be examined, including administration, application, and platform areas.
- Completion of this workflow **eliminated a customer-visible call outage window** during RSM setup, a mandatory requirement. This design specification was the first in the JWR release to cross all development groups, and it set a new standard for cooperation between groups.

**Systems Architect, Platform Group:** This effort involved an architecture analysis to identify key failure points and bottlenecks of DMN data flow to improve call load performance in JWR.

- Initiated the idea for this architecture analysis effort and worked closely with the JWR platform group to quickly integrate results concurrent with analysis. This focused the performance testing and enhancement efforts to rapidly zero in on suspect bottlenecks and processes.
- Mapped industry standard design patterns to existing codebase to derive a component architecture model of the system.
- Utilized a pipe and filter model to find bottlenecks and enable measurement of system resource usage and process behavior at boundary limits of all operations.



- Completion of this analysis and integration of the results eliminated several customer-visible system lockups that were occurring under peak-hour loads.

**Troubleshooting, Real-Time Systems:**

- Provided on-call support to field engineers in Japan to isolate defects and recommend workarounds involving the DMN and RSM products.
- Worked closely with Development QA to design and perform test cases, reproduce defects, trace defect causes, and implement fixes.

**Bell South Subscriber Migration Project**

**Software Engineer, Platform Group:** This effort required a port of the Subscriber Migration (SM) product from the Bell South legacy codebase to the Bell South COA GR0.3 project. All coding done using gnu C for SCO UNIX and Univel targets under SunOS ClearCase.

The SM component supported the movement of subscribers between co-located platforms, and was the ancestor of the Remote Subscriber Migration (RSM) product worked on in previous assignments.

- Reverse engineered workflow and supporting processes from hundreds of pages of obsolete documentation and legacy code. Added support in underlying code to migrate subscriber features added in COA GR0.3.
- Supervised QA personnel to test and revise workflow in progress.
- Completion of this effort **reduced subscriber service interruption during SM by over 75%, streamlined a 5-day operational process to 3 days, and condensed field operations documentation by over 75%.**

**Technical Writing:**

- Filled customer requirement to revise legacy workflow documentation from a nested hierarchy of hundreds of steps to a simple linear progression optimized for field engineers. Documentation changed from practically unusable to elegant.

**Troubleshooting, Real-Time Systems:**

- Worked closely with Development QA to design and perform test cases, trace defect causes, and implement fixes, with customer present in lab.
- Assisted customer to reproduce field environment and perform customer acceptance testing in lab.
- Supported field engineers during initial live deployment.

**Bell Atlantic Remote Subscriber Migration Project**

**Systems Architect, Platform Group:** This effort required a new design of the migration engine for the Remote Subscriber Migration (RSM) component of the Bell Atlantic ANP GR0.2 project. All coding done using gnu C and C++ for SCO UNIX and Univel targets under SunOS ClearCase.

The RSM component supported real-time migration of subscribers between networked platforms, typically for movement to a new system release, without interruption of service. This process was transparent to the subscriber, triggered on-demand through subscriber service requests or queued by a background daemon.

- Worked with platform group RSM team to analyze and design solutions for cross-platform subscriber database translation needs, network protocol enhancements, integration issues surrounding destination platform real-time subscriber database updates, and failed operation recovery issues.

**Software Engineer, Platform Group:**

- As developer of the core migration engine, a table-driven finite state machine, implemented engine routines and the underlying library code needed for real-time cross-platform database record translation.
- Assisted in the design of user interfaces and their integration with the migration engine.

Strategic Leader • Critical Thinker • Hands-On Coder • Agile Architect • Continuous Idea Generator

- Refactored all network protocol code in applications and daemons across product codebases.
- Completion resulted in a **new shared codebase** with a core migration engine that could be easily extended across projects. This went on to become the **baseline RSM code for all future projects** in the company.

**Technical Writing:**

- Set new standards for analysis and design documentation at the company during rollout of ISO 9000 compliance. Coordinated team efforts on these documents.
- Created the largest, most accurate, and most useful functional and design specifications at the company.

**Process Engineering:**

- Performed exhaustive post-mortem analysis of all major defects in RSM for traceability and specific recommendations for methodology improvement.

**Troubleshooting, Real-Time Systems:**

- Worked closely with QA on RSM field systems to trace defect causes, and implement fixes.
- Key member of an emergency bugfix team for Digital Messaging Network (DMN) component across multiple project codebases.
- Key resource for tackling bugs at remote customer sites and labs.

**CONSULTING SOFTWARE ENGINEER / Financial Instruments Database Services**

*Fidelity Investments, Vantage 20/20 / Boston, MA*

**September 1995 - September 1996**

The Vantage 20/20 project to synchronize legacy brokerage systems to a **UNIX / Oracle** platform had been blocked by data integration issues for a year. As analyst and developer, resolved core issues through the redesign of the broker data model, data house-holding processes, and daily round-trip database synchronization. Set new standards for virtually 100% code coverage, while piloting **Oracle** Designer.

**Business Analysis:** Identified and analyzed all Fidelity business processes that affected the NCIP database.

- Worked with Fidelity business units to develop business process models using **Oracle** Designer\*2000.
- Identified legacy broker data quality issues, and advised responsible business units to take corrective action.
- Created NCIP data control and validation process for NCIP datasets supporting all Vantage projects.
- Designed process to synchronize broker data between relational and non-relational (legacy) systems.
- **Resolved fundamental data issues that had stalled the NCIP Broker release since October 1994.**

**Software Engineering:** Created client-server code to support the database models and business processes developed in earlier stages of the NCIP project for deployment in the Common Order Processing (COP) Engine release. Coded in Lucid C and **Oracle** Pro\*C under SunOS ClearCase and Fidelity Summit-D methodology.

- Developed and tested client-server code for **Oracle** connected Tuxedo servers to support Windows and **UNIX** clients. This code enabled access to the NCIP database, and the synchronization of broker information between Vantage (relational) and legacy (non-relational) systems.
- Demonstrated ability to perform concurrent research in CASE tools while delivering projects on schedule.
- Developed Tuxedo API coding techniques to **achieve virtually 100% code coverage** for the release. This level of coverage had been heretofore unknown and was achieved in record time.
- This release **achieved closure on the broker data phase of the project, which had previously failed.**

kentdorsey@gmail.com

# Kent Dorsey

(617) 606-5184

Strategic Leader • Critical Thinker • Hands-On Coder • Agile Architect • Continuous Idea Generator

**Database Analysis:** Modeled broker data for Non-Customer Involved Party (NCIP) project, part of largest Oracle database ever constructed at the time.

- Created entity-relationship models for NCIP using Bachman Groundworks and **Oracle** Designer\*2000.
- Supported multiple releases of the NCIP data model and version controlled the underlying dataset.

## Additional Career Experience

Please visit <http://linkedin.com/in/kentdorsey> or <mailto:kentdorsey@gmail.com> for information on additional career experience.