Alexey Omelchenko

Phone: 512-900-9513
E-mail: Alexey.S.Omelchenko@gmail.com

SENIOR SOFTWARE DEVELOPER

SUMMARY
- Over 18 years of software development
- Application performance analysis and optimization
- Design and development of multi-tier enterprise level systems
- Excellent C++ with emphasis on object oriented programming
- Cross-platform development and porting from one platform to another
- Comprehensive knowledge of data modeling on physical and logical levels
- Over 10 years in major financial institutions in USA and Europe
- Understanding of banking, investment, and financial services industry needs and transactions
- Excellent problem solving skills
- Excellent communication and interpersonal skills, leadership and teamwork

SKILLS
Programming languages: C++ (C++14), C#, .Net, C, SQL, Python, PowerShell, Assembler
Libraries & technologies: STL, Boost, Posix, Win32 API, MFC, ATL, WTL, COM, Intel TBB, FIX, Tibco RV, ZeroC ICE, ZeroMQ, MQTT, Solace
Web technologies: Rest, SOAP, HTML, XML, XSLT, JSON
NoSQL: Redis, Aerospike, MongoDB
RDBMS: Oracle Database, MS SQL Server, MySQL, Sybase ASE
Software platforms: Linux, Solaris, Windows
Development tools: Git, Bamboo, Bitbucket, CVS, SVN, Clearcase, Source Safe, GCC, ICC, Visual Studio, Code::Blocks, PyCharm
Performance tools: Intel C++ Compiler, Intel Performance Libraries, Intel VTune
Hardware platforms: x86, x86-64 (AMD64), XScale, Itanium
Virtualization: VMware, Xen, OpenVZ, VirtualBox
Cloud Platforms: PCF
Development process: Agile, Iterative methodologies, Scrum, TDD, ATDD, FDD, XP

EXPERIENCE
2016 – Present, Lead Software Developer, Charles Schwab (schwab.com), Austin, TX
Systems: Enterprise Data Bus, Market Data and Quotes Gateways, Document Flow Automation
A set of middleware services and protocols that provide various financial and markets information to front office and client facing applications. The EDB is a high performance, secure and robust transport solution for data streaming across multiple systems inside the company and for external client applications. It is built on top of MQTT protocol.
- Participated in development of Enterprise Data Bus - a complex solution built around Solace messaging system and MQTT protocol.
- Implemented core component of the solution - Subscription Manager.
- Implemented LDAP protocol and LDAP server – another core component.
- Implemented plug-ins API, SDK and debugging tools for the plug-ins development.
- Analyzed performance and scalability. Identified bottlenecks and formed a list of strategies to eliminate them.
- Improved performance by factor of at least 100.
- Assumed the role of an interface for external teams, consulted them about development of their applications, discussed business and technical requirements for the data bus.

- Worked with Solace to add new authentication features to their product, e.g. OAuth token authentication.
- Participated in defining the minimum viable product with the business departments.
- Worked with architecture team and local governance to get approval for the proposed solution.
- Mentored my team about tools and technologies necessary to build the reliable, secure, fast and scalable system.
- Migrated several services to the new messaging solution as a proof of concept.
- Implemented multiple RESTful WebAPI services that provide data for Schwab web site.
- Supported ~30 legacy applications written in Java, C++ and C#. Retired some of them.
- Supervised secure coding compliance for the department of ~50 people, ~80 applications.
- Used formal Scrum methodology.

Environment: C++, C#, Powershell, MS SQL Server, Redis, Aerospike, TFS, Git, JIRA, Confluence, Bamboo, Bitbucket, Windows, Linux, OAuth, SAML, MQTT.

2013 – 2016, Software Engineer, NCBI (www.ncbi.nlm.nih.gov), Washington D.C. Metro Area
System: NCBI Authentication System
A Federated Authentication System which integrates authentication systems from hundreds of universities all over US as well as NASA, Google, PayPal and others. Core component of the system is deployed in similar organizations in Canada, UK and Korea. The system is cross-platform. It can be built for Linux and Windows. NCBI runs it on Linux servers, foreign institutions – on Windows. The system processes 20 million transactions per week.
- Assumed responsibility for the authentication system development and engineering support.
- Communicated to dozens of institutes' IT services to establish new or repair/upgrade existing integrations with NCBI authentication.
- Worked with multiple scientific and engineering teams to implement NCBI authentication for their projects.
- Redesigned Database Layout to improve performance of the system.
- Improved reliability of the system through analysis of performance bottlenecks, memory management issues, opportunity for data caching.
- Implemented several modules from scratch.
- Worked with UK, Canada and Korea teams to adapt their installation of the system and help with data migration.
- Implemented automated monitoring and statistics reporting, ensured the system stability.

Environment: C++, Python, Perl, STL, Boost, .Net, MS SQL Server, MySQL, SVN, Git, gSOAP, Pthreads, gcrypt, OpenSSL, Jira, OAuth, SAML, CGI, FastCGI, Windows, Linux

2010 – 2013, Lead of Development Team, Otkritie (www.open.ru/en/), Moscow, Russia
Systems: Internet Banking, SugarCRM, OTRS, CreditRegistry, Accounting System, SMS Information Service
A set of systems designed for needs of Bank's corporate clients and VIP oriented business. Different systems are integrated. Some services are written from zero, another modified and adapted from open source projects.
- Headed development and integration of in-house information systems and open source solutions
- Participated in integration of an external Internet Banking System and our internal Accounting System – determined format and data structure of documents, UML
- Participated in customization of SugarCRM to internal needs – set of integrations and custom improvements for the systems: Internet Banking, Accounting System, Analytic System and United Data Storage
- Implemented real time performance monitoring for business critical systems using testing framework and monitoring system, Nagios
- Obtained solid knowledge of corporate banking services

Environment: C++, C#, Python, Perl, PHP, STL, Boost, Oracle, MS SQL Server, MySQL, SVN,
            ZeroC ICE, NUnit, CppUnit, UML, Redmine, Windows, Linux

2009 – 2010, Senior Software Developer (AVP), Deutsche Bank ([db.com](db.com)), Moscow, Russia
Systems: ARIS, Score
Two automated risk management systems that valuate more than 1M trades daily. They are written in C#, use external risk engine written in C++, running on DataSynapse grid. DBMS – Oracle. For some types of reports FireBird is used.
They are deployed to 2 Citrix boxes, 10 application servers and pool of 3k DataSynapse engines.
- Supervised system development as technical owner and lead developer
- Delivered 4 major releases of the systems without issues
- Coordinated work of development team in Moscow and business in London, Singapore and South America
- Obtained knowledge of valuation of asset derivatives – Foreign Exchange, Money Market, FI, IR Swaps
Environment: C#, C++, STL, Boost, Python, Oracle, DataSynapse, SVN, UML, Jira, Confluence,
            Windows

2007 – 2009, Senior Software Developer, UBS ([ubs.com](ubs.com)), Moscow, Russia
System: Termi
Solution designed for securities trading, pricing, order management, trade audit and much more. It is used for trading of different derivatives. Termi is composed of a large number of processes; a huge number of batch jobs, scripts and cron jobs; and at least 3 databases.
- Transferred ownership of the system from another external team in 1.5 months
- Supported the system until its total decommission
- Assisted in step-by-step migration of dataflow into another new systems
- Maintained legacy code in C, C++, Objective C, Perl, C Shell
- Obtained knowledge of pricing of FI, FX derivatives
Environment: C++, C, Objective C, Perl, FIX, TCP/IP, Socket, Crontab, UML, Oracle, Sybase,
            MS SQL Server, C Shell, Jira, Confluence, Solaris
System: Ultrabond
Solution for Mortgage-Backed securities data analysis designed as client-server software. Client side has Excel-like interface with tables, figures, charts and support of data import and export to common formats. Platform: C#, Windows. Server side is distributed scalable computing system with single front-end and multiple back-ends, written in C++ for Solaris and Linux. Communication layer is Tibco Rendezvous and ZeroC ICE.
- Integrated the system with new data feeds – an external database, Oracle
- Developed Perl script for conversion and import of historical data form a set of files
- Ported server side software for Windows – the code became cross-platform – Solaris, Linux, Windows
- Obtained knowledge of valuation of fixed income and in particular Mortgage-Backed securities
Environment: C++, C#, Perl, STL, Boost, Threading, Syncfutsion, Tibco RV, ZeroC ICE, Sybase,
            Oracle, CppUnit, NUnit, Clearcase, UML, Jira, Confluence Linux, Windows, Solaris

2006 – 2007, Senior Software Developer, CQG ([cqg.com](cqg.com)), Moscow, Russia
Product: CQG for Windows

Solution for all professional trader needs. It includes market coverage for futures, options, fixed income, foreign exchange, equities and indices. It incorporates several different interfaces – depth-of-market order book views, charts, spreadsheets.

- Guided development team of 4 people as a team lead
- Performed performance analysis of the application, found several sources performance bottlenecks.
- Designed event based data provisioning for multithreaded user interface.
- Implemented set of features as a custom ActiveX control.
- Obtained knowledge of financial market terminology and business processes

Environment: C++, COM, ActiveX, STL, Boost, WTL, MFC, CVS, UML, Windows

2004 – 2006, Software Developer, ABBYY (abbyy.com), Moscow, Russia
Product: ABBYY Recognition Server (http://www.abbyy.com/recognition_server/)
Server-based OCR solution for automated document processing designed for high volume workload with transparent scalability and fault tolerance.

- Designed and implemented GUI management console for the server
- Improved scalability for huge-size multipage documents processing
- Ported OCR Engine from Windows to Linux
- Released first version of the Recognition Server

Environment: C++, STL, MFC, ATL, COM, RPC, MMC, Windows Service, VSS, UML, Windows, Linux

2001 – 2004, Software Developer, Intel (intel.com), Nizhny Novgorod, Russia
Product: Intel Performance Libraries (http://software.intel.com/en-us/articles/intel-ipp/)
Highly optimized cross-platform software functions for multimedia, data processing, and communications applications.

- Obtained outstanding experience in software performance analysis and optimization
- Implemented and highly optimized diverse functions for Intel processors: Pentium 4, Itanium, XScale
- Implemented algorithms for matrix processing, video processing, cryptography, big number arithmetic, XML processing
- Performed competitive performance analysis of MySQL engine built with GCC vs. Intel C++ Compiler. That was done for joint project of Intel Compiler team & MySQL
- Designed several libraries including cryptography and XML processing
- Acquired deep knowledge of cross-platform software development
- Obtained important experience of working in a global organization and performing to its corporate values
- Honored with a Very Special Thanks Award

Environment: C, C++, C#, Java, P/Invoke, JNI, Assembler, VSS, CVS, StarTeam, BitKeeper, NUnit, JUnit, Rational Purify, Intel VTune, UML, Windows, Linux, PVSC Tracker, Agile, XP

EDUCATION & SERTIFICATIONS

2008 – C++ Programming, International Knowledge Measurement, Score 99%
2001 – MS in Physics, Moscow Institute of Engineering Physics, Russia

EXCELLENT REFERENCES AVAILABLE UPON REQUEST