

David S. Bakin
Bakin's Bits, LLC

davidbak@gmail.com

<http://www.bakins-bits.com>

1602 E Garfield St C
Seattle, WA 98112
(206) 724-4395

TECHNICAL STRENGTHS

Design and development of concurrent (multithreaded) programs for multicore processors. Theory of concurrent systems. Debugging of concurrent systems. Low-level concurrency primitives and lock-free data structures.

Microsoft Windows XP tools and application, client-server, middleware, and distributed application design and development. Microsoft development technologies including .NET, VC++, ATL. Structured data technologies including XML, XSLT. Microsoft component technologies including automation, controls, COM.

Object-oriented design and development. Object-oriented architectures for applications and systems. Class and Object Framework architecture and development. Functional programming design and development.

Expert in C++, C; very experienced in C#, x86 assembly. F#. Java. Compiler & development tool development.

WORK EXPERIENCE

Amazon.com, LLC

2009—2011

Fulfillment by Amazon

Implemented the "Amazon Fulfilled Shipments Report", for all FBA merchants, customized for all global regions. Implemented a data pipeline to move FBA ordering data from production databases through the data warehouse into reporting databases, combining and cleaning all shipment item data on the way. Optimized the queries for efficiency during the daily runs. Implemented the report using the FBA reporting framework. Validated all data pipelines, and report accuracy. (Thousands of these reports are pulled daily.) Subsequently implemented a separate data pipeline for China invoicing data. Implemented a new "freshness" mechanism (for all FBA reports) to show merchant, in the reporting UI, how recently the data was updated, reducing merchant calls for support. (SQL – Oracle dialect, Java, Perl)

Providing training, design reviews, and design support for the entire FBA team for data warehouse issues, including best practices for development production data warehouse applications to reduce operational overhead, and correct and efficient use of SQL to take advantage of (and not misuse) the special operational behaviors of the data warehouse.

Satori Software

2008—2009

Mailroom Toolkit

Analyzed US and Australian Post Office rule changes and implemented them in the address correction engines. (C++)

Mailroom Capture

Implemented a keystroke capture facility so that user keystrokes could be displayed to user in a “symbolic” form, and then replayed into an arbitrary application at a later time. This involved capturing individual keystrokes and using a hidden window to discover how Windows would interpret key sequences including Alt-*nnn*, Alt-0*nnn*, and Alt+*nnn*. (C#, C++)

Microsoft

2006—2007

Photosynth (<http://photosynth.net>)

Wrapped low-level concurrency primitives. Provided an implementation of condition variables. Provided debugging features specific to the concurrency patterns used in the Seadragon Engine. Provided a protected work queue and task pool for the master/multiple-slave threading pattern. Diagnosed and fixed concurrency and asynchronous operation issues involving data races, deadlocks, and performance. (C++)

Implemented the “point cloud” 3D model view in Photosynth. Implemented an octree-based compression scheme for the point cloud raw data to speed point cloud loading from the internet. Implemented asynchronous loading and incremental display of the point cloud so that displaying the model’s point cloud did not interfere with the interactive performance of the image view. Optimized the point cloud display for efficient use of GPU memory (under DirectX 9) (as well as main memory). (C++)

Implemented frame-rate performance tracking, and frame-rate metering to achieve smooth interactive performance of the image view under varying load conditions. (C++)

Seadragon

2005—2006

Seadragon Phone Client

Ported Seadragon client code to Windows Mobile 2003 Pocket PC Second Edition Phone Edition. Replaced the application’s outer layer with Pocket PC specific code. Replaced the application’s low-level drawing methods with Pocket PC’s GAPI. Made a native Win32 implementation of the application’s abstract concurrency classes. (C++)

(Seadragon was acquired by Microsoft in February, 2006.)

Google Inc.

2003—2005

Google Pack (<http://pack.google.com>)

Designed and coded the mainline operations of the application, and was responsible for all security design and implementation of this security-critical application. Designed and implemented a module for querying a Windows system for the presence or absence of third-party applications – applications attributes were described in XML – used this to integrate multiple third-party applications into this application, tested it against 10 third-party applications. Designed and implemented a “hook” mechanism for tracking progress of (third-party) child processes, and monitoring changes to the registry and file system. Created the install, auto-update, and uninstall of the application. (C++)

Google Desktop Search (<http://desktop.google.com>)

Created from scratch the install, auto-update, and uninstall of the application, including OS-embedded components (BHO, deskbar, Layered Service Provider) – key attributes: reliability, small overall size, and speed. Implemented reliable multi-threaded components for asynchronous internet access to files and other services. Implemented many “foundation” classes providing access to various Windows technologies and services. Contributed to the performance analysis and optimization of on-disk data structures. (C++)

Microsoft Corporation

1997—2003

Productization of Attributed Call Profiler (SDE, 3 person team total, 1 yr)

Productized the client-side instrumentation, on 5 architectures, of an attributed call profiler to be reliable and fast, so that the product could be shipped. Redesignated and reimplemented callstack tracking (including support for Win32 and C++ exceptions, fibers, and multiple calling conventions), wrote assembly language instrumentation for Pentium procedure call/return interception and timing, and created “fast paths” for common cases. (C++)

Feasibility Investigations (VMWare, Retail POS) (SDE, sole responsibility, 1mo VMWare, 3Mo Retail POS/each)

Performed independent evaluations of strategic choices for the Windows CE Tools group.

- 1) Evaluated alternatives for a Windows CE emulator by: Established requirements. Investigated commercial choices (incl. VMWare and Connectix). Prototyped a Windows CE emulator on the commercial platforms. Planned product development for each alternative. Measured performance. Recommendation to not use VMWare was accepted.
- 2) Investigated specific functional needs to enhance Windows CE penetration into the Retail POS market. Compared acceptance of various OPOS versions, and OPOS/OPOS.NET future direction. Presented results in two white papers. Recommendation to support (newly formed) MS Retail POS division in proposed turnkey cash register replacement project was accepted.

DirectShow Performance Improvement (PM, sole responsibility for perf, ~2 yr)

Contributed (as Program Manager) to the performance improvement of DirectShow. Initiated development of performance regression test suites and their weekly run and analysis. Initiated source code analysis effort to find and fix performance problems in display and capture of video. Presented at ST-IF3 conference for Microsoft. Original work led to three patent applications.

Living Room Computing ('Pandora') (PM, sole responsibility for reliability, ~1½ yr)

Contributed (as Program Manager) to specification and prototype development of an adaptation of Windows 98 to be an "easy to use, no maintenance" platform in the living-room environment. Investigated operating system and application changes needed for a self-maintaining system. Initiated a probe of user-unfriendly error messages. Developed a mechanism to log and correlate "interesting" events in applications and the operating system to lead to an operating system which could self-diagnose and self-correct for error situations. Developed performance techniques for doing streaming media operations on a limited hardware platform (single spindle). Original work led to two patent applications.

Intrinsa Corporation

1996—1997

Prefix for Windows Development (SDE)

Contributed to the development and completion of the Prefix for Windows product, a development tool using software simulation technology to detect defects in source code. Modeled Windows and C++ APIs for use in simulation, including MFC and STL. Developed techniques for detecting leaks and other misuse of Windows resources during simulation. (C++)

Sybase, Inc./Powersoft Corporation

1994—1996

Internet-Enabling OLE Control Toolkit (Dev mgr, 4 person team, ~6 mo)

Development Manager on a project to develop an Internet toolkit for client-server programmers. This suite of cooperating OLE Controls (ActiveX Controls) allowed programmers to easily access internet services and build internet applications. Designed and implemented an Object framework that provided user-level networking functionality over TCP including persistence, reliability (in the face of intermittent net failure), automatic retry, delayed transfer via deadline scheduling, logging, name service, and caching. Controls included file transfer (ftp/http), HTML munging (parsing, editing), URL munging (parsing, relations), Browser control and management, and Server interface (nsapi/isapi/cgi) with state management. (C++)

Agent-Based Distributed Workflow Automation (Dev lead, 3 person team, ~9 mo)

Development manager on a project to develop an Agent-Based Distributed Workflow Automation System. Managed the project team, including determining the functional specification of the product, resources, and schedule. Proposed the project. Architected the distributed O-O Workflow Automation System on top of an agent execution infrastructure. Defined layered COM approach for end-user and administrative UI components. Defined COM based transport for workflow cases. Built Object/Agent framework and workflow-specific agents for routing, tracking, and role assignment, and foundation agents for logging and data storage. Used standard technologies including RPC, ODBC, MFC, COM. Negotiated requirements with other Sybase project teams. (C++)

Windows Client/Server Development Tool: Build Momentum (SDE, ~3 mo)

Conducted a performance analysis and Added performance-driven improvements to Build Momentum (database client app development environment). Added UI classes and events to the Framework Class Library. Replaced 4GL UI objects with native Windows UI controls for improved performance.

Microsoft Corporation**1992—1993****Windows NT System Development, Windows 3.1 Application Development (SDE)**

Responsible for design and development of filter/query UI O-O fully-extensible Windows Shell for next-generation Windows NT (Cairo). Developed SQL and File System OLE monikers, managing access to MS SQL Server and ODBC data sources. Developed query UI OLE components.

Responsible for GUI authoring environment of a database-centric workflow groupware application, which included a graphical editor for process flows and an IDE for scripting workflow actions.

Software Publishing Corporation**1990—1992****Windows 3.1 Application: InfoAlliance for Windows (SDE)**

Ported the GUI front end of the InfoAlliance application, written using SPC's platform independent GUI layer, from Presentation Manager to Windows 3.1. Responsible for system issues including front end performance under Windows, memory management and segmentation, compatibility with other Windows applications, and Windows-specific issues. Maintained the GUI database design tool.

Office Products Technology**1989—1990****Windows 2.x/3.0 Application: Formation Business Form Design Package. (SDE)**

Designed and developed a fully GUI electronic business-forms design and printing application for Microsoft Windows, including a rich-text control.

Rational**1986—1989****Rational Environment: Ada Cross-Compiler Development (SDE)**

Developed Rational's retargetable middle-end as member of cross-compiler development team. Performed retarget of entire middle-end to M1750A architecture. Designed the retarget to the IBM 370 for VM/CMS and MVS

PATENTS AWARDED

7,058,947	Resource Manager Architecture Utilizing a Policy Manager	June 6 2006
6,799,208	Resource Manager Architecture	Sept 28 2004
6,763,439	Disk Throttling and Priority Queuing for Streaming I/O	July 13 2004
6,748,443	Unenforced Allocation of Disk and CPU Bandwidth for Streaming I/O	June 8 2004

PATENTS PENDING

Method and System for Effective Management of Client and Server Processes

May 2002

Resource Manager Architecture with Resource Allocation Utilizing Priority-Based Preemption

May 2000

EDUCATION

Boston University

Boston, MA

M.S., Computer Science

Harvey Mudd College

Claremont, CA

B.S., Mathematics