Phillip Markovic

Address

FAKE ADDRESS

Telephone Number

FAKE NUMBER

FAKE NUMBER Home

FAKE NUMBER Mobile

Email

FAKE EMAIL

Tertiary Education

Completed the Bachelor of Science (Computing Science) at the University of Technology, Sydney in Spring 2000 with a **Credit Average rating.**

In 1993 commenced study in the Bachelor of Arts in Economics - Finance Studies at Macquarie University. Completed one semester successfully.

Secondary Education

Completed the Higher School Certificate at Marcellin College Randwick in 1992. The Tertiary Entrance Rank obtained was 81.60.

Employment History

Kaz Group Limited

August 2003 – Present

POSITION HELD: Analyst Programmer

At Kaz I have worked for the Speech and Technology Group in a number of areas. These areas are focused on voice recognition and speaker verification technology. Virtually all of my work has been C++ programming tasks with some scripting and Java development.

Speech FX Voice Recognition Platform
I was involved in the following tasks on the speech recognition platform:

- Developed and maintained automated testing software.
- Developed test cases for the automated testing environment.
- Development work on a tape backup and restoration system. Each night the entire source code repository was backed up to tape. It was also restored to another location to ensure the integrity of the backup. The tape containing the backup is taken off site each day.
- Documented the Speech FX front end. The front end is a component of the speech recognition platform which pre processed the raw samples from a speech data stream. It transformed the data into a form, which could be processed by the recogniser component. I documented the algorithms used in each stage of the transformational pipeline.

- Developed logging software (Network Logger). This allowed the platform to write its logs to another machine across a network. It relieved the machine hosting the speech platform from having to manage extensive numbers of log files.
- Assisted in developing the build infrastructure.
- Maintenance of the build infrastructure.
- Enhanced existing tools including:
    - SmProcWatcher – Server software used to start programs on the host machine. I turned this into a windows service.
    - SmProcClient – This would issue commands to the SmProcWatcher. I modified this to process extra commands.
    - SmTestWave – This runs a separate instance of the platform to simulate a real call over the telephone network. I added the ability for two instances of Speech FX to do out of band communications in addition to normal telephone communication.
- Restructured Perforce repository tree and created a new third party location. This was to allow all branches of the source code to link to the third party libraries using the same relative path information.
- Migrated Perforce from Win32 to a Linux operating environment.
- Rebuilt the build machine (the server containing the automated build and test environment).
- Completed programming work on an application called VPIMServer. This was the backend to one of our voice applications, which allowed a caller to check their voice messages using voice commands rather than key press options. This application would manage the messages other callers left and would play them back upon request.
- Created temperature monitoring software for our server room. This program would periodically send a http request to a thermometer in the server room, which was connected to the network. If the temperature returned was above a certain threshold, this program would initiate shutdown of all machines in the room in a specific order. This included the UPS hardware. This application has a client and server component. I developed it to be built and run on both Windows and Linux as we have machines using both platforms in the server room.
- Maintenance of Speech FX. Any issues that were logged in our bug tracking system and assigned to me.
- Developed a utility for displaying the contents of Front End Grammar file. The front end of the Speech FX platform did the initial signal processing of raw speech data. Many required parameters were taken from the grammar file to perform this task. The grammar file follows a particular binary format. This tool allowed a developer to easily view and understand the contents of this file. It would also specify which components used the various grammar file values.
- Developed a file dependency utility for extracting whole portions of the Speech FX source tree. This allowed various parts of the platform to be compiled and tested in isolation. It also permitted a novice to learn the structure of the platform at a faster rate.
- Developed a utility for renaming entire directory trees. This is a

general purpose tool, which was created as part of my effort to move our source repository from a Win32 to Linux environment. Under Linux, Perforce source control systems followed different directory naming conventions. Owing to the enormous number of directories that needed to be renamed, I thought it would be better to automate the process.

- The machine housing the build infrastructure contains a RAM drive. This drive contained all the third party libraries and source code required for the nightly build. This drive was used to speed up the nightly build process. Some dependencies had to reside on this drive at all times. However, because this drive was volatile, a reboot of this machine would wipe its contents. To counter this I developed a script based utility for restoring the lost contents onto the RAM drive upon system restart.
- Providing support for Perforce within my group in correspondence with the Perforce company helpdesk.

Speaker Verification

We refactored the entire Speech FX platform and enabled it to process Voice over IP. I was assigned to develop and maintain two components, these being the Prompt Player and the Verifier.

- The Prompt Player component would read raw voice data from a particular source. This could be a telephony stream or a file. It was responsible for introducing voice data into the platform and would regulate the rate at which this data was fed into the system.
- The Verifier was used to establish a caller’s identity based upon the vocal characteristics of that persons voice. These are stored in a voice print file. The verifier component was also responsible for creating these voice prints through an enrolment process.
- Created and updated the design documents for each of these components. It involved establishing use cases and other UML diagramming.
- Modified the Verifier component so that it could read and write voice prints to a database. I developed classes to write binary data to MySQL and IBM DB2 databases. These classes are generic enough to be used in other software projects.
- Modified the Verifier component further so that it could encrypt and decrypt voice prints.
- Modified the Application layer component to allow it to interact with the Windows Active Directory Service using the LDAP protocol. This allowed access to usernames on a company wide network.
- Modified the application layer component to include extra capabilities. For example, I added a feature which allows callers to use a telephone or mobile phone keypad to spell out a username rather than saying it.

Comsec Updates and Maintenance

As part of my regular duties I have also performed ongoing updates and maintenance of an existing application for Commonwealth securities called Voicebroker. It allows callers to do share trading and listen to stock quotes.

- I was required to add recognition for new company names and codes every few weeks. This was an involved process as new company names and codes had to be compiled into the grammar. Deletion and modification of existing company details were often required as well. I also had to record company names so they could be played back to the caller.
- This system has been in existence for a number of years. During those years its popularity increased to the point where call recordings and log files took up too much hard disk space. Once a month I log onto these machines using my PCAnywhere client to remove these files and free up disk space as necessary.

TOOLS AND TECHNOLOGY USED:

- Microsoft Visual C++ 6.0
- Microsoft Visual C++ 2003
- Perforce (Source Control)
- Enterprise Architect (Designing Software Artefacts)
- Active State Perl (Primary Scripting Language)
- Adaptive Communications Environment (ACE – A C++ framework for concurrent communication software tasks. The original implementation of our speech platform utilised this heavily.)
- Java Software Development Kit.
- MySQL and IBM DB2 using ODBC.
- Goldwave (Software for creating, manipulating and recording sounds and saving them as different sound file formats).
- PCAnywhere
- Windows XP Professional
- Windows Server 2003
- Linux Fedora Core Versions
- GNU Tools and the Linux development environment

VeCommerce Limited

April 2001 – June 2003

POSITION HELD: Software Engineer

While at VeCommerce I was involved in a number of projects as part of a team. The work undertaken in each project consisted of a number of tasks for which the internal VPMM methodology was employed, and analysis, design and supporting documentation was produced.

Ladbrokes (UK)

VeCommerce developed a sophisticated NLSR system to take bets for this London betting agency. As part of the project team I developed two applications in C++ using MFC to allow customers to tune horse and venue names for text-to-speech and speech recognition.

Dialect (News Limited)

As part of the team for this project, I developed the callflow and host interface in C++. This NLSR system is generic and allows callers to pay their bills with a credit card using natural speech.

Pizza Hut

I designed and implemented the callflow application and host interface. The host interface was written in C++ using proprietary libraries to interact with the VeCommerce technology platform. The host interface communicates with a backend AS400 and PABX using TCP/IP and draws configuration information from a MySQL database.

I created a tool using C++ for parsing trace logs generated by the VeCommerce platform.

I created another tool using C++ for generating regular reports about the stage the caller reached for each call placed into the system.

TAB Ltd (NSW TAB)

VeCommerce developed a speech recognition system for the NSW TAB. I made changes to the existing bet engine. This application was written in C++ and contains most of the callflow logic. I designed and implemented new modules for this application so that comprehensive betting information could be logged for each call placed into the system. I also made changes to the callflow. The information generated by the new logger was used to produce statistical reports on a regular basis.

I designed and implemented a client utility in C++ to load test the bet engine.

I designed and implemented software in C++ to eliminate performance overheads imposed by the system having to process thousands of logs.

Australian Taxation Office

I worked on the IVR for the Australian Taxation Office. I updated a number of existing callflows and created new ones upon demand. I also created the host interface, which allowed the callflow to communicate with the backend host. This was done in java. The host interface also connected to an access database for configuration information. I also created a host interface emulator for testing any new applications developed for the ATO at the office before deployment.

Internal Development Work

Between project work I was involved in developing tools to be used internally. I developed compact host/host interface emulation software that is completely configurable. I was also involved in creating a platform emulator so that developers could test applications on their desktop rather than doing scheduled tests using shared resources. This was done in C++ as well.

I worked on the VeCommerce auto attendant, which allows external callers to be connected to an employee by stating their name.

TOOLS AND TECHNOLOGY USED:

- Microsoft Visual Studio (Visual C++)
- Microsoft Visual Source Safe (Source Control)
- Microsoft Visio 2000 (Design Diagrams)
- Java Software Development Kit.
- Velocis, MySQL, SQL Server and Microsoft Access DBMS using ODBC.
- VeCommerce First Contact Platform (Integrates voice recognition, text-to-speech and telephony resources).
- VeProfile (Web based System Monitoring Tool).
- VeCreate (Visual RAD tool for creating callflow applications).
- Cool Edit Pro (Software for creating, manipulating wave forms and saving them as different sound file formats).
- Windows NT/2000.

Aymcrest Pty. Ltd.

February 1999 – March 2001

February 1994 – December 1997

POSITION HELD: Computer Support /Data Entry Operator and Dairy Produce Merchant

DUTIES:

- Set up, support and maintain a new computer system required to achieve business objectives. This operation is comprised of multiple PCs that act as backups for each other, should one fail another PC can take over.
- A commercial software package is used and I have been involved in training users and setting up the system.
- I have also been involved in training other users on various PC applications and supporting computing problems.
- On a daily basis I perform data entry tasks, printing of summaries and various types of billing statements. I also process / journalise all business accounting transactions.
- Product distribution to wholesale customers and government institutions.

TOOLS AND TECHNOLOGY USED:

- Windows based PC running Supervend. This is a commercial package developed for milk vendors. It is used to process daily orders and for accounting purposes.
- Supervend is a Paradox database with a Delphi front end.

Access Systems Research Pty. Ltd.

January 1998 – January 1999

POSITION HELD: Undergraduate Software Engineer

Access Systems Research was focused on developing online gaming systems for organisations with a gaming license.

I developed an online timesheet system. This was implemented as a java applet connected to an access database through ODBC.

I developed and maintained issue tracker, an online system for tracking software bugs. This was done using Perl CGI programming, javascript and HTML. This was also interfaced to an Access database through ODBC.

I developed and maintained an online inventory tracking system for tracking the whereabouts of items like laptops, books, software and hardware items. This way we would always know who had an item last. People were required to use the system each time they borrowed an item.

I was also responsible for maintaining and regularly updating the RSVP.com website which is an online dating service. It was in its infancy at the time and was operated as a separate venture out of the same office.

I was also involved in the development of the companies testing tools, which interpreted a markup language developed internally. This testing suite automated testing of the front end, middleware, security layers, and backend. It enabled thousands of transactions to be performed overnight for instance. This was done entirely in Java, and other than for the front end was run mainly under Unix.

TOOLS AND TECHNOLOGY USED:

- Java Software Development Kit.
- Web Development: Perl/CGI, HTML, Javascript.
- Windows NT Server with Internet Information Server.
- ODBC data sources.
- Microsoft Access 97 databases.
- SQL.
- UNIX commands and utilities.

R.C. White & Co.

November 1992 - May 1993

POSITION HELD: Clerk
DUTIES:

- I processed / journalised business transactions
- I prepared bank reconciliation statements
- I performed word processing and spreadsheet tasks
- Data processing, mail collection and banking were also other tasks I undertook.

TOOLS AND TECHNOLOGY USED:

- PC running DOS with accounting software.

Computer Skills

Programming Languages

C/C++
Java
JavaScript
HTML
Perl/CGI Programming
SQL (Structured Query Language)

UNIX commands and utilities
MSDOS commands and batch file programming

Operating Systems

Windows Server 2003/XP Professional/ 98/ NT/ 2000
MSDOS
Unix - SUN / SPARC workstation
Linux

Other Software

Adaptive Communications Environment
Perforce
Microsoft Visual Source Safe (Source Control)
Enterprise Architect
Microsoft SQL Server 7.0 (Desktop Edition)
DB2, MySQL, Velocis and Microsoft Access using ODBC
Oracle SQL Plus for Window 98 (Limited exposure)
Goldwave
PCAnywhere

Microsoft Visio 2000
Microsoft Excel
Microsoft Windows
Microsoft Word

Other Skills

Fluent in verbal and written Spanish.

Current MR class NSW Drivers License.

Interests

Software Development

Reading IT and science journals

Swimming

Rugby League

Boxing

Student Activities

A member of the school debating team.

A member of the school mock trial team.

A member of the school public speaking team.

Selected for the Commonwealth Bank Cup school boy Rugby League competition.

Awards Received

Awarded a Team Excellence award for outstanding contributions and accomplishments on the Pizza Hut project.

Nominated on two occasions for the company’s monthly exceed award.

Awarded Credit merits in Mathematics and Science competitions.

Various subject awards throughout academic history.

Referees

MR. PETER CHIN
Chief Technology Officer
KAZ
Telephone (02) 9808 0303
Mobile 0413 757 690

References

MS. MEGAN WHITLOCKE
Project Manager
VECOMMERCE LIMITED

MR. BRETT FELDON
Software Engineering Manager
VECOMMERCE LIMITED

DR. PETER TOWSON
Chartered Professional Engineer
PG & JE TOWSON
CONSULTING CIVIL AND MECHANICAL ENGINEERS

MR. RENE CHARLES WHITE
Chartered Accountant
R.C. WHITE & CO.

MR. PAUL DEVINE
Managing Director
ANCHOR INVESTIGATIONS
INSURANCE & PRIVATE INVESTIGATIONS SURVEILLANCE -
CORPORATE SERVICES