**Terrence Monroe Brannon**

**Summary:**

Over 15 years of professional IT experience. Over 9 years of Python development experience. Strong emphasis on software quality, using these coding ethics:

* Separation of concerns to leverage functional composition (think: “S.I.C.P.”)
* Holistic code strengthening as opposed to “bugfixing by band-aids”.
* Modular development, documentation and testing of application business logic

**Education:**

* M.S. Computational Neuroscience, University of Southern California, May 1999. Thesis research: Non-linear synaptic integration in cortical pyramidal neurons.
* M.S. Computer Science, Lehigh University, Dec 1993. Thesis: Chameleon: An Embedded Language Perspective for Multiple Computer Language Programming.
* B.A. Computer Science, Morehouse College, May 1991.

**Skills:**

* Python, SQL (Postgres, Oracle, SQLite and MySQL), Ab Initio, HTML, XML, and Linux.

**Experience:**

**Oppenheimer Funds, Denver, CO Oct 2018 – Mar 2019**

**Consultant**

Tools used: Python, SQLAlchemy, Reahl, AWS

Skills used: Web application development, Object-relational modeling.

* Designed a relational data model for check reconciliation. Also created scripts to seed the database with data, relational views and stored procedures. Modeled the relational schema in the SQLAlchemy ORM.
* Developed an object-oriented graphical user interface to the object-relational check reconciliation data model. Use of Reahl eliminated the need to write javascript, html and css, allowing to focus on connecting the high-level object-relational model to high-level graphical widgets.
* Developed deployment scripts to re-package the app for deployment to AWS.

**Metalevel Consulting, Atlanta, GA Jan 2017 – Oct 2018**

**Consultant**

Tools used: Python

Skills used: ETL, API consumption, algorithmic trading, cryptocurrency, marketing.

* Developed ETL scripts in Python to reconcile financial records of the VivaCoin cryptocurrency project. Project involved querying and summarizing data from the Steemit blockchain.
* Developed algorithmic cryptocurrency trading programs in Python for a Canadian firm.
* Sole developer of SurgeTraderBot, a Python-based FOSS algorithmic trading bot that trades cryptocurrency and provides daily reports of results.
* Developed training documents for acquiring and investing in cryptocurrency.
* Streamlined the bulk email process for a small business funding company leading to a 5x jump in efficiency.

**Bank of America, New York, NY Jun 2016 – Dec 2016**

**Python Developer**

Tools used: Python (Quartz), FIX Protocol. Jira

Skills used: Object-oriented programming, Unit testing, ETL, event-driven programming.

Participated in the entire software development lifecycle (software development, support, requirements gathering, release management) for a large graphical desktop application that captured internal and external credit derivatives trades. The application numbered in the hundreds of thousands of lines of heavily object-oriented Python code. It was used, developed and supported around the clock by a team of developers spanning 3 time zones on a 2-week agile release cycle. The application was event driven and made use of object-oriented databases, message queues and relational databases.

Refactored an existing application to reconcile trade data between various trade storage sources (message queues, RDBMS, OODBMS, etc.). Existing application failed due to uniprocessor time/space overload. Parallelized memory intensive parts. Most proud of the polymorphic report classes which took the transformed and analyzed data and incrementally collated and built up the output reports in a number of formats.

* Implemented widgets to specialize the behavior of the graphical user interface as a function of the product type being traded. Did so in a completely clean manner as opposed to the gaggle of if-thens and implicit types previously used for the same purpose.
* Added new business trade logic and insured that the trade data was supported at data and visual level in all aspects of the main application (searching, matching, rendering, validation, storage, retrieval, etc.) as well as migration and reconciliation of the data.
* Assumed ownership of a large project that facilitated the exercise of index options.

This position had a number of strong points (thorough agile development, integrated development environment, cross-team support and education) while suffering from some serious Achilles' heels (code smells, lack of story points for support roles, infidelity in staging environments, use of database in unit tests).

**General Electric Healthcare, Remote Jun 2015 – Dec 2015**

**Software Developer Engineer in Test (SDET)**

Tools used: Python (behave, pyautogui, argh), Windows, Perforce

Skills used: Object-oriented programming, Automated testing, behavior-driven development (BDD).

* Developed behavior driven development code in Python to automate several phases of software testing in an Application Lifecycle Management tool. Test automation included automating GUI interactions as well as consuming and developing REST APIs. Developed feature files and implemented the feature file specifications in Python using the behave Behavior-driven development tool.

**Cisco, San Jose, CA Jan 2015 – Jun 2015**

**Python Technology Lead**

Tools used: Python (Nagare and SQLAlchemy), Red Hat Enterprise Linux (RHEL), Subversion

Skills used: Object-oriented programming, Web application development.

* Developed a web user interface for a database-driven parsing engine. The parsing engine was responsible for parsing free form text files whose content was stereotyped into several content styles. These content styles were modeled as a hierarchy of business objects and mixins. The business objects leveraged polymorphism for stepwise rendering and validation of the user interface as well as stepwise creation and storage of new database-backed parser types.

The entire project was written in Python, with SQLAlchemy forming an object-relational interface to the database and Nagare forming an object-relational interface to rendering and validation of complex web forms.

**AIG, New York, NY Jul 2014 – Dec 2014**

**Software migration and release control**

Tools used: Python, Red Hat Enterprise Linux (RHEL)

Skills used: Software migration and release control.

* ETL Python programs to estimate claims were ported from Windows to AIX and Redhat Grid servers. Abstracted the filesystem using Python singleton objects as a domain-specific language and wrapped the native file operations leading to a highly DWIM inline code body.
* Developed Python scripts to archive and copy directories between machines.

**Google, Cambridge, MA Oct 2013 – Mar 2014**

**Software Development Engineer in Test (SDET)**

Tools used: Python (Selenium), Ubuntu Linux

Skills used: User Interface testing

* Developed software to test the user interface for the Google Play for Education Project. Abstracted pagination for reuse across multiple parts of the site using iterators. Refactored authentication for improved readability.

**Bank of America, Jersey City, NJ Sep 2012 – Mar 2013**

**ETL and GUI Developer**

Tools used: Python and Quartz (a proprietary event-driven object-oriented database)

Skills used: ETL, object-oriented programming

* Developed stateful GUI components to record job details. Converted job details to a pipeline of ETL SQL queries. Reused the SQL generation for the GUI for batch SQL jobs. Prototyped job control dependency using Quartz's dependency tracking.

**JP Morgan Chase, New York, NY Jul 2012 – Sep 2012**

**Middle-Tier Developer**

Tools used: Python and Athena (a proprietary object-oriented event-driven Python datastore)

Skills used: business process management

* Participated in a 5-week proof-of-concept project. Participated in providing 1 of 4 candidate solutions to replace the existing business process management tool.
* Implemented the event-driven middleware between a graphical user interface and the Java and PL/SQL business operations. The middleware was responsible for responding to graphical job initiation by forwarding requests to the business layer. It polled the business layer for results and communicated them to the GUI. Implemented decorators to improve the re-use of code to trap and report exceptional behavior.

**Truveris.com, New York, NY Feb 2012 – May 2012**

**LAMP Developer**

Tools used: Python (pylons, SQLAlchemy), Postgres, Javascript (mooTools), PyCharm

Skills used: web development, database design

* The work in this group was very routine. It consisted of adding a new feature to a web application by migrating the database schema and providing a new URL to map the new functionality to. Some of the heavily interactive front-end functionality exposed a serious design question: how to synchronize data models when the front-end and back-end language differ.

**BiotrackTHC.com, Fort Lauderdale, FL Apr 2011 – Feb 2012**

**ETL and GUI Developer**

Tools used: Perl (Mojolicious), gtk, Postgres, jQuery, Subversion (svn)

Skills used: gui development, web development, data munging

* Database design and development
* Developed GUI and programmatic interface to the Quickbooks SDK from Perl
* Generated Excel spreadsheets detailing the transport of products

**Verio.com, Boca Raton, FL Jan 2010 – Jun 2010**

**Data Engineer**

Tools used: Perl (DBIx: Class, Moose), Oracle, Sybase

Skills used: SQL, data modeling/analysis, object-oriented programming

* Specialized in customer billing in the Ordering, Provisioning and Billing group
* Wrote SQL, DML and DDL to improve relational integrity and provide business logic
* Simplified release control and DML generation via scripts
* XML-RPC API development and debugging

**Ohio Board of Regents, Columbus, OH Sep 2009 – Dec 2009**

**Data Engineer**

Tools used: Perl (Sybperl) and Sybase

Skills used: SQL, object-oriented programming, web programming

* Wrote and modified SQL to retrieve and display student financial aid eligibility data. Because of the sensitive nature of the functionality, the program was rigorously tested.

**InsuranceAgents.com, Columbus, OH Jan 2009 – Sep 2009**

**Developer**

Tools used: Perl (Moose, CGI: Application, DBIx: Simple), MySQL, XML, git

Skills used: ETL, object-oriented programming, web programming

* Developed a concurrent lead vending system. Created XML data feeds from relational data and submitted the data to multiple partners for bidding.
* Co-developed a Perl/CGI website for intranet CRM and reporting/billing.

**JP Morgan Chase, Columbus, OH May 2007 – Dec 2008**

**ETL Engineer**

Tools used: Python, Perl, Ab Initio, MS SQL Server, DB Visualizer, Weka

Skills used: ETL, object-oriented programming, parsing, machine learning

* Replaced Perl programs designed to detect money-laundering with Ab Initio graphs.
* Built a Python/Perl record linkage system to do using fuzzy matching and parsing.
* Used DB Visualizer to analyze stored procedures and convert to Ab Initio graphs.

**Innovaro, Tampa, FL Oct 2006 – May 2007**

**Data Engineer**

Tools used: Python (pycurl, pyparsing, ftputil), Perl, Apache, CGI, MySQL, MS SQL Server

Skills used: ETL, object-oriented programming, web programming

* Performed in a data processing role spanning several Innovaro web properties:
* KnowledgeExpress.com - implemented a configurable suite of object-oriented Python and Perl data processing programs. These programs used heterogeneous access and parsing methods to regularize a jungle of vendor data for commit it to our MS-SQL database for uniform front-end search.
* Pharma-Transfer.com - wrote a Python program to email a newsletter of recent relevant search results to each customer.
* TechEx.com – performed database and CGI program maintenance as well as some front-end web design (with a DOM approach to HTML templating).

**Freelance Consulting, Southern California and New York City Sep 2002 – Mar 2006**

**LAMP and Data Engineer**

Tools used: Perl, MySQL, Sybase

Skills used: ETL, web programming, object-oriented-programming.

* Vendare Media - Dynamic HTML generation using DOM tree rewrites (as-opposed to ASP-style inline programming).
* Valueclick - Full stack LAMP development.
* Directsynergy - ETL Data Engineering
* City of New York - convert Excel spreadsheets to XML.

**Oracle, Redwood Shores, CA Jan 2001 – Nov 2001**

**Senior Perl Developer**

Tools used: Perl (DBI, Net: FTPServer, LWP), Oracle, PL/SQL, HTML

Skills used: web programming, SQL, object-oriented programming

Worked in the Applications Integration group. This group is responsible for releasing software patches and complete distributions via a database-driven mod\_perl web site.

* Authored perltest.pl, which generated and formatted the questions for the Oracle India Perl Competency test. Also responsible for more than 50% of the actual questions used.
* Sole developer of ARUFTPD, a virtual file-system FTP server. This server was derived from the CPAN Net: FTPServer to provide navigation and retrieval of files and directories on a remote file system with an FTP front end. Authentication and authorization were handled via SQL queries on an Oracle database.
* Used libwww-perl to write a "web spider" to simulate browser navigation across multiple screens (including login and authentication) in order to allow command-line based download of patch files. Initial implementation visited several pages in succession, parsing each page for its action to decide on the next page to request. The redesigned version subclasses the existing download class with a batch derivation and the resulting LWP script made one-page call and the derived class called all the screens by calling the Perl methods to generate each screen. In eyes, initial approach was better, but higher management preferred the latter. Wrote a regression test suite to ensure usage under normal and exceptional cases.
* Developed a mod\_perl handler, which allowed for partial-content (resumable) downloads.
* Wrote a PL/SQL program which scanned all Oracle product families for those with upload responsibilities and added these responsibilities to the appropriate administrators.

**Freelance Consulting, Southern California and New York City May 1999 – Dec 2000**

**LAMP and Data Engineer**

Tools used: Perl (Net: FTP, DBIx: Recordset, Cache: Cache, CGI/mod\_perl)

Skills used: Object-oriented programming, CGI, Database driven application, ETL

* mcy.com (Manhattan, NY) - Developed scripts and modules to parse and generate fixed-width text files for various EDI (electronic data interchange) feeds. Extended Text: FixedLength with Text: FixedLength: Extra by creating a simpler API and adding support for leading zero and floating-point formatting.
* Instinet/Reuters - Developed INETCROSSFTP, an FTP system with the following phases: drag and drop GUI interface, file conversion, FTP file upload/download/monitoring, and PGP-based file encryption. The program used a 3-level hierarchy of object-oriented behaviors. In retrospect, one level of protoyped-based objects (or Moose roles) would have been equally powerful. The base class was HostCycle. All subclasses cycled across FTP servers performing their until () method until it returned true at which time, they executed their then () method. Abstracting common FTP usages led to development of Net: FTP: Common for CPAN.
* angryman.com (Queens, NY) - Developed a persistent distributed framework in which Vote objects were committed to a File: Cache and then resurrected and committed to database via slave objects. Used Date: Manip to develop Date: Horoscope, a module that determines Zodiac sign based on birth date (These days Date: Range might be more appropriate). Also used Date: Manip to determine a person’s age bracket based on birthdate.
* end70.com (Irvine, CA) - Used Date: Manip to realistically populate a MySQL customer database with lease initiation and shipment dates. Developed a CGI interface for SQL temporal queries of a database. Wrote DBI scripts to randomly allocate client leads to salesmen based on various time criteria
* hosting.com (North Hollywood, CA) - Developed a set of Perl, CGI, and Javascript scripts as well as a MySQL database to collate and analyze flat file user execution logs from 300 web servers. The phases of the project were:
* Designed a multi-table MySQL database in which to store log entries.
* Wrote a Perl/DBI script to parse Apache logs and store them in the created database.
* Wrote a Perl/DBI script to query the database. Queries made of use table joins. Complex queries were handled by creating temporary tables.
* Developed a web interface to the Perl/CGI query script.