



# Tom H Anderson

Lead PHP Architect

## Profile

This narrative describes my work through the lens of a single project spanning 25 years that demonstrates the breadth and variety of my experience: [etreedb.org](http://etreedb.org).

1998, [etree.org](http://etree.org) pioneered the distribution of bit-perfect live concert recordings over the Internet. I built the database to keep track of every live concert in circulation, and to let site users add those recordings to personalized lists. I chose **PHP 3** for scripting and **MYSQL** for storage. Having already been a professional database developer in **FoxPro**, **PowerBuilder**, **Oracle**, **Sybase** and **MSSQL**, good database design came naturally. PHP and MYSQL were new, especially because PHP lacked an **IDE**.

Multi-user support came in December 1999 thanks to **PHPLIB**. Site users can submit corrections and new data freely. By June 2000 I had created tools for users to administer this data with full auditing. **I now have over 300 such volunteers**.

The idea of community-generated content that anyone could improve, lightly moderated, was huge. It predicted and predated **Wikipedia** (started 1/15/01). [etreedb.org](http://etreedb.org) gives every user a profile and allows them to curate the content listed there. The self-publication concept of putting your list online was a forerunner of **MySpace**, **Facebook**, and similar.

Quickly, I faced numerous complications due to the popularity of the site. **Hits peaked at 20M per month**. PHP frameworks were still in their infancy. I started using **Smarty** templates and wrote custom **cache** handling. I wrote one of the first public PHP bootstrap files, `set_env.php`. Then **BitTorrent** was created and the etree community was the first user. [archive.org](http://archive.org) volunteered its resources to store etree collected recordings.

To help manage its growth, **ibiblio** volunteered to host the application, and [etreedb.org](http://etreedb.org) remains **the largest site hosted by this public library**.

With site stability ensured, I wanted to create a way for anyone to query the site in any way they wanted. I tried many different approaches beginning in **2011** with the integration of **Zend Framework 1**.

Soon I was invited to join the **Zend** team building **Doctrine** integration for **Apigility** featuring **Hypertext Application Language**. I created [api.etreedb.org](http://api.etreedb.org): an attempt to redesign the database with **Zend Framework 2**. The API worked but could not run the website. It was the idea of an API-driven tool that anyone could access to query etreedb data that I was after.

I wrote `api-skeletons/zf-doctrine-graphql` and tried using that. I joined the **Doctrine open-source team**. I moved from **Laminas** to **Laravel** and

## Details

Millcreek, UT

USA

(559) 341-5098

[tom.h.anderson@gmail.com](mailto:tom.h.anderson@gmail.com)

## Links

[tomhanderson.com](http://tomhanderson.com)

[etreedb.org](http://etreedb.org)

[LDOG Stack](http://ldogstack.com)

[graphql.lcdb.org](http://graphql.lcdb.org)

## Skills

PHP

MySQL

GraphQL

Doctrine ORM (team member)

Laravel (contributor)

Skipper ERD (leading expert)

Google Cloud Platform

Speaker

Eagle Scout

Scores of Successful Projects  
Across the PHP Landscape

Contributed to Over 150  
Open-source Projects

Expert with Github

Exceptional with Docker

## Certifications

Zend Certified Framework  
Engineer, and PHP  
Programmer.

wrote many open-source libraries for Laravel and Doctrine. I decided what I needed was a GraphQL database on top of etreedb.org's existing database without any changes to the database structure.

I aliased the whole database with **Skipper** and now feature [graphql.lcdb.org](http://graphql.lcdb.org), which is built from my template application [ldog.apiskeletons.dev](http://ldog.apiskeletons.dev). This site completes all my dreams of a resource anyone can query for any doctrine supported database using the superior libraries of Laravel, Doctrine, and GraphQL in PHP. It is called the **LDOG Stack**.

## Employment History

### **CEO at API Skeletons, Inc., Millcreek, UT**

June 2015 — Present

Proprietor and programmer specializing in PHP GraphQL API database applications. Worked with over 80 clients through Upwork with a 100% success rate. Projects range from a few hours to several years.

### **Senior PHP Programmer at Wolfe, Pittsburgh, PA**

September 2023 — February 2024

Member of core applications team using Symfony and Doctrine ORM. I started the Developer Experience Guild to form recommendations for improving code quality across all programming teams.

### **API Architect at NoteVault, San Diego, CA**

February 2016 — September 2017

The company was using open-source libraries I created with Zend and Apigility. Led the development of an Elastic Search implementation that transformed the performance of NoteVault applications.

### **API Architect at Lodgable, Midvale, UT**

July 2015 — January 2016

Created an API integration between Lodgable and a content provider with over 100 endpoints. I left when the integration was done, and I was later told that Lodgable never had to change my integration.

### **Sr. PHP Programmer at Soliant Consulting, San Mateo**

March 2011 — July 2014

As a consultant, I worked with clients of Soliant Consulting on various projects. We started using Zend Framework 2 at Beta 3.

### **PHP Programmer at Power Music, Salt Lake City, UT**

March 2009 — June 2010

Created a complex music track editing tool with jQuery. Created an API to combine sales data across all properties.

### **Other Experience at Miscellaneous, Salt Lake City, UT**

May 1995 — March 2008

Professional programming beginning 1995. Available upon request.

### **Utah Army National Guard at US Army, Salt Lake City, UT**

May 1990 — May 1998

91Q10 - Pharmacy Technician