
EXPERIENCE

Information Technology Services, University of Michigan

Ann Arbor, MI

MiWorkspace Engineering

DevOps Architect Supervisor

2017–Present

Izzy as a Service To expand the scope of Mac management to all of campus, I proposed and led an initiative to sell Izzy, the Ruby platform I developed to control `munki` and Jamf, as an independent service. I met with unit leadership, technical stakeholders, and end-users to understand their needs and discuss the benefits of embracing Mac management technologies. I provided pilots, training, and ongoing support during each unit's full-scale roll-out process. I coordinated bringing on all but one campus department, school, and college – from dozens to thousands of systems – including Michigan Medicine (PHI), the College of Engineering (ITAR), Ross Business School, and the Dearborn campus. As of today, we have just over 10,000 systems under management. I have worked with Michigan Medicine on splitting their VMware Workspace One environment to only provide management for BYOD systems. For my work on Izzy, I received the President's Staff Award of Distinction in 2019.

Izzy Client In preparation for Apple Silicon systems, I rewrote our library of on-client scripts and tools – many in Ruby or Bash – into a single unified Swift binary. This frees us from relying on a scripting runtime or installing one on each client. Additionally enables TCC controls for full disk access and exceptions from EDR tools.

Izzy at Home When COVID-19 hit, I was able to scale up our emergency-use zero-touch install method to full production within a week, and we were primarily using zero-touch by the end of March 2020. I unified our bootstrap process to work for either on-campus or at-home deployments, and we continue to use zero-touch with about 1 in 4 systems. We also use zero-touch in situations where a system needs to be returned to service and it is inconvenient for the customer to bring it to campus. I worked with our Windows team to help build an offline domain join registry in Go for their InTune-based at-home deployment.

Additional Tooling I wrote and deployed an API server – written in Swift Vapor, deployed to Kubernetes – for our block-M menubar tool. I created a few automations in Go (and deployed to Kubernetes) to validate our Jamf database, and then notify us in Slack – for example, when a duplicate system is registered after a logic board replacement.

Telemetry and Monitoring In addition to the reporting Izzy and Jamf both gather, I plumbed in Elasticsearch, providing robust dashboards and metrics to monitor the state of the overall fleet. We used these dashboards to adjust deployment deadlines for macOS patch updates, realizing that the majority of users upgraded within a week of release.

Plenary Developer I have been a lead developer on tools for collaborative services, including a Microsoft Teams creation and synchronization engine to Azure ActiveDirectory, a pre-flight validator for Slack Enterprise migrations, and GitHub Enterprise internal tooling (all in Ruby). I am a member of the Technology Review Board which all new projects must meet with before developing solutions. I have been an early adopter of cloud, devops, and container technologies, and advise on and advocate for modern devops practices.

Mac Team Lead

2015–2017

MiWorkspace With the roll-out of Michigan's NextGen's desktop service – MiWorkspace – I became the team lead, overseeing four direct reports. I was responsible for coordinating work between the team and each University department as they enrolled in the service, meeting with technical specialists in each unit, and adjusting our services to meet their specific unit's needs. During this roll-out phase, we simultaneously had four different units in different states of engagement, from preparatory meetings, proof of concept, actual nightly deployments, and post-deployment adjustments.

NoMAD Login In order to move off Active Directory binding, I created a custom mechanism inside NoMAD Login in Objective-C to address our specific needs around SecureToken handling and administrative access. Our team also migrated from NoMAD to the Single-Sign-On extension in macOS Big Sur.

DevOps I transitioned Izzy's infrastructure to running first in Docker and managed with Ansible, then finally to a managed Kubernetes cluster in AWS; in addition to continuous deployment, automatic resiliency reduced on-call demands. I designed and deployed an on-campus package content delivery network, delivering terabytes of package updates weekly to thousands of clients.

Jamf With increased requirements around DEP, supervision, and MDM profile deployments, we deployed Jamf Pro to support initial bootstrap and profile management. We continue to use `munki`'s best-in-class package resolution. Jamf is controlled via APIs called from the Izzy web app.

PaperCut Follow-Me iOS Job Release I wrote a Swift iOS app to let customers release "Follow-Me" print jobs. I used Bluetooth beacons to identify which printers are nearby, falling back to scanning a QR code sticker on the printer. A later version added an iOS sharing acceptor to let users directly print PDFs to the follow-me queue from their iPhone or iPad.

Platform Senior Engineer

2012–2015

NextGen Michigan In 2012, the University redesigned central IT services, including – for the first time ever – a full desktop management service. Izzy became the key technology for Mac management, and I was responsible for implementing Izzy at campus scale. We scaled to over 4,000 clients by 2014.

Automation As part of NextGen, we hired two additional Mac team members. Together, we expanded my Jenkins-based package automation to around 300 different titles, all automatically picked up and deployed to test and QA clients, and then deployed to production weekly. I was the lead developer on the automation pipeline, helping my teammates develop techniques in HTML scraping and command-line package creation.

Iron Izzy I designed a FileVault management toolchain for Izzy, called Iron Izzy, to enable FileVault and key escrow, for Mac OS 10.7 clients. The tool was built in Ruby and Objective-C, and coordinated key rotation with Izzy. All MacBooks built with Izzy were additionally secured by Iron Izzy.

Block M Menubar I wrote a menubar tool in Objective-C for quick access to support links and shared storage. It received its configuration from Izzy, and could be customized at a business unit level.

IzzyStor I wrote a Mac OS app to archive and restore systems for rebuilding them – an offline migration assistant. These tool was used by the night onboarding team to archive, reload, and restore up to fifty systems a night. For safety, IzzyStor encrypts all archives at rest.

Campus Computing I coordinated the replacement of `radmind` to Izzy for the University's public use computing sites, approximately 500 clients across 60 locations.

College of Literature, Sciences & Arts, University of Michigan

Ann Arbor, MI

LSA Mac Lead

2010–2013

Simple NetInstall Managed the College's Mac deployment system, updating it for Mac OS X 10.6 and new packages every semester. This system was retired when Izzy launched alongside Mac OS X 10.7

Izzy With a shoestring budget, using `munki`, `Rails`, and `packagemaker`, I created a management system for Mac OS at the college, called Izzy. Izzy lets the support staff in the field deploy new packages to systems at any time through a simple web interface, rather than at build time with NetInstall or by hand, speeding installation request resolution times. Izzy allows IT staff to manage defaults for their builds, and each department's access was siloed. Izzy ramped to over 1,000 systems in its first five months of service.

Package Automation Automated some very common packages (Firefox, Adobe Flash Player, Chrome, etc) using Jenkins. This allowed the college to be extremely timely with Flash security updates, deploying updates within hours of the new version being released, and requiring no manual steps.

IzzyBoot Built a bare-metal imaging workflow for Macs, by creating an Objective-C wrapper around `asr`. IzzyBoot leveraged a decision engine in Izzy to pick which base OS was appropriate for the device, as Mac OS versions were occasionally model-specific. IzzyBoot registered the client with Izzy and pre-deployed `munki`, making builds faster and more reliable than installing by hand.

BlueReview Wrote an end-to-end lecture capture service, backed by `Rails` and Mac OS technologies `AVKit` and `EventKit`. This service captured over 100 lectures a week from the 12 largest auditoria on campus.

Life Sciences Institute, University of Michigan

Ann Arbor, MI

System Administrator

2003–2010

Infrastructure Designed and implemented the entire computing infrastructure of a new institute from the ground up, including all-new switchgear, multiple Xserve RAIDs in a SAN, an Xserve-based compute cluster, Open Directory on Mac OS X Server, VMware vSphere, Active Directory, SCCM, and Exchange.

Automated Genome Mining Coded a cluster work scheduling system for bulk, automated genome mining, based on a tool developed at the Institute. doi:10.1186/1471-2105-10-185.

Desktop Engineering Created a Mac OS build system based on NetInstall and some scripts, making builds repeatable, easy to manage, and simple for student staff to perform. Implemented a "golden-triangle" configuration with Active Directory and Open Directory. Became a Certified Mac Technician and formed an AppleCare SSO, speeding up our turn-around time for repairs. Provided repair services to nearby University units. Implemented SCCM for Windows desktops.

Purchasing System Created an institute-wide purchasing app to streamline and centralize ordering, reducing inefficiencies and allowing researchers to focus on science.

IT Contract Services, University of Michigan

Ann Arbor, MI

System Administrator

2000–2003

Supported Solaris and RedHat Linux customers across campus. Provided security consulting and incident remediation services.

Surveys of Consumers, University of Michigan

Ann Arbor, MI

Web Developer

1996–Present

Develop and manage the Surveys web properties, including internal-, public- and sponsor-facing data distribution sites and tools. Primarily implemented in PHP and Ruby on Rails.

EDUCATION

University of Michigan

Bachelor of Science in Engineering, Computer Engineering *with honors*

Ann Arbor, MI

1996–2000