Taking Control of Your Clinical Trial: Running ZZedc Independently

A guide for investigators who want to own and operate their own electronic data capture system

Clinical Research Technology

2025-12-07

Table of contents

1 Introduction

The landscape of clinical research data management is changing. Investigators increasingly want **full control** over their data, systems, and operational decisions. Yet many commonly used electronic data capture (EDC) systems create **vendor dependency**: expensive licenses, proprietary data formats, and the uncomfortable reality that your system operates at the pleasure of your vendor.

This post documents **ZZedc**, an open-source, investigator-owned EDC platform designed with a different philosophy. ZZedc runs on infrastructure you control, uses standard databases and formats, costs a fraction of commercial EDC systems, and—most importantly—can be deployed and managed by your research team without ongoing dependency on any biostatistics lab or commercial vendor.

1.1 What This Post Covers

- **The independence problem:** Why control matters in clinical research
- **ZZedc overview:** What it is and how it differs from commercial EDC systems
- **Deployment paths:** From a single investigator's laptop to multi-site trials on AWS
- **Technical architecture:** How ZZedc achieves simplicity and security
- **Getting started:** Step-by-step guides for different deployment scenarios
- **Long-term sustainability:** Ongoing maintenance, migration, and data ownership

---

2 The Independence Problem

2.1 Why Investigator Control Matters

Clinical researchers routinely face uncomfortable situations with EDC systems:

Vendor dependency: Your study relies on continued vendor support. If the company changes pricing, goes out of business, or decides to discontinue support, you're stuck.

Data ownership ambiguity: Proprietary data formats mean your data isn't truly "yours." Exporting data can be expensive, slow, or impossible without vendor cooperation.

Operational constraints: You can't customize validation rules, reports, or workflows without vendor professional services (and associated costs).

Security concerns: Your patient data sits on vendor infrastructure. You don't control security, backup locations, or data residency.

Cost escalation: EDC licensing grows with patient volume. A system that costs $20K/year at baseline often costs $50K+ by study completion.

2.2 Common Scenarios Where Investigators Break Free

In practice, these situations drive investigators to seek alternatives:

Collaboration breakup: A productive collaboration with a biostatistics lab ends due to cost disagreements or service level mismatches. The investigator is left with data trapped in a vendor system.

Study expansion: A pilot study is successful and scales to multi-site. Existing EDC licensing becomes prohibitively expensive. The investigator needs a more scalable approach.

Regulatory concerns: The investigator's institution requires data to be stored in a specific location or under specific security controls that the vendor can't accommodate.

Long-term stewardship: The study becomes long-term follow-up. Vendor relationship doesn't last 5+ years, but the investigator must maintain the system and data indefinitely.

Methodological evolution: As analysis evolves, the investigator needs flexible validation rules, custom reports, and integration with analysis tools. The vendor system feels restrictive.

2.3 Existing Solutions Have Limitations

Solution	Advantage	Limitation
Commercial EDC (REDCap, Medidata)	Polished interface, vendor support	High cost, vendor lock-in, data ownership questions
Spreadsheets (Excel, Google Sheets)	Familiar, free	No validation, poor audit trail, compliance issues
Homebrew databases (Access, FileMaker)	Customizable	No security, poor scalability, compliance nightmare
Custom R/Shiny apps	Completely flexible	Requires skilled programmer, no pre-built features
ZZedc	**Open source, low cost, independent**	**Requires basic technical setup**

---

3 Introducing ZZedc

ZZedc is a **modern, open-source electronic data capture system** built with R and Shiny. It's specifically designed for investigator ownership and independence.

3.1 Core Design Philosophy

Investigator-centered: Every design decision prioritizes your control. You own the data, the system, and the infrastructure.

Cloud-native but independent: Deploy on AWS, Azure, Google Cloud, or local servers. You choose the infrastructure provider based on cost and compliance needs.

Security and compliance by default: GDPR and 21 CFR Part 11 compliance frameworks built in. Data encryption, audit trails, and electronic signatures are standard features.

Open source, not proprietary: Source code is available on GitHub. If you need customization, you can do it yourself or hire any consultant—you're not locked into the vendor.

Standards-based: SQLite databases, YAML configuration, standard web technologies. If you need to migrate away, your data is in standard formats.

3.2 What ZZedc Includes

Electronic data capture with real-time validation: Enter study data with immediate field-level validation. No waiting for data manager review.

Role-based access control: Five built-in roles (Admin, PI, Coordinator, Data Manager, Monitor) with configurable permissions.

Comprehensive reporting: Basic enrollment reports, quality control summaries, and statistical overviews—all built-in, no custom programming.

Data quality framework: Automated checks for missing data, outliers, and consistency across visits. Nightly QC runs identify issues early.

Audit trail and compliance: Every action is logged with user, timestamp, and change history. Electronic signatures supported for regulatory studies.

Data export and analysis: Export to CSV, Excel, SPSS, or R. Integrate with your preferred analysis tool directly.

User-friendly admin interface: Create users, manage backups, view audit logs—all from the web interface. No command-line expertise required.

---

4 How ZZedc Achieves Investigator Independence

4.1 Architecture: Simplicity and Transparency

```
(Authentication, Forms, Reporting)

        ↓ Database

SQLite Database (Standard Format)
Your Data, Your Control
```

Key design points:

- **All infrastructure is standard**: Docker containers, standard web server (Caddy), open-source database (SQLite)
- **No proprietary components**: You're not dependent on any vendor-specific technology
- **Transparent processes**: You can see, audit, and modify every part of the system
- **Portable data**: Your data is in SQLite—you can access it with any SQL tool, analyze it with any tool, migrate it anywhere

4.2 Deployment Paths: Flexibility for Different Needs

ZZedc supports multiple deployment approaches, depending on your scale and technical resources:

4.2.1 Path 1: Solo Researcher (Local Laptop)

Best for: Individual investigators, small pilot studies, initial prototyping

Install on your personal laptop in 10 minutes:

```r
# 1. Install ZZedc package from CRAN
install.packages("zzedc")

# 2. Initialize project with interactive setup
library(zzedc)
zzedc::init()  # Answers 15 simple questions

# 3. Launch
launch_zzedc()
# Application opens in browser at http://localhost:3838
```

Data lives in a local SQLite file (`data/zzedc.db`). You're the only user. Backup by copying the file to cloud storage.

Cost: Free (only your laptop electricity) **Maintenance**: Minimal (you run it when needed)

4.2.2 Path 2: Team Research (Single AWS Server)

Best for: Research team at single institution, collaborative multi-site trial

Deploy on AWS EC2 in ~15 minutes:

```
# 1. Install AWS CLI, configure credentials

# 2. Run deployment script
./aws_setup.sh \
  --region us-west-2 \
  --study-name "Depression Treatment Trial" \
  --study-id "DEPR-2025-001" \
  --admin-password "SecurePass123!" \
  --domain trial.example.org \
  --instance-type t3.medium
```

Application runs on AWS infrastructure. Multiple team members access via HTTPS. Database backed up automatically.

Cost: ~$30-50/month for EC2 instance (or less, depending on size) **Maintenance**: Basic (Docker handles updates, Caddy handles HTTPS renewal)

4.2.3 Path 3: Enterprise/Multi-Site

Best for: Large NIH-funded studies, pharmaceutical trials, production deployments

Deploy across multiple AWS availability zones with load balancing, RDS database, S3 backup, CloudWatch monitoring.

Cost: $200-500/month depending on data volume **Maintenance**: Automated (infrastructure-as-code, CI/CD pipeline)

4.3 Security and Compliance

ZZedc includes security and compliance frameworks that commercial EDC systems charge extra for:

GDPR Compliance: - Data subject rights portal (access your data, request deletion) - Purpose limitation (users only see data they need) - Audit trail of all access

21 CFR Part 11 (FDA): - Electronic signatures with role-based authorization - Immutable audit trail with hash chaining - System validation framework

Security Baseline: - Password encryption with configurable salt - HTTPS with automatic Let's Encrypt certificates - Role-based access control - Session timeout and concurrent login limits

---

5 Deployment Scenarios and Examples

5.1 Scenario 1: Solo Researcher Prototype

Dr. Jane is a clinical psychologist planning a small depression treatment study. She wants to test intervention feasibility before seeking NIH funding.

Her approach:

```
# Install on her laptop
install.packages("zzedc")
library(zzedc)

# Quick setup (5 minutes of questions)
zzedc::init()
# - Study name: "Depression CBT Pilot"
# - Target enrollment: 20
# - Admin username: jane_smith
# - Password: (secure password)

# Launch
launch_zzedc()
# App opens at http://localhost:3838
```

Result: Jane has a secure, validated EDC system running locally. She can: - Create forms for baseline, weekly, and endpoint visits - Enroll patients and enter data - Generate enrollment reports - Export data to Excel for analysis - Back up by copying a single file to Dropbox

Cost: $0 **Timeline**: 15 minutes from zero to collecting data

---

5.2 Scenario 2: Multi-Site Trial Migration

The ADHD research consortium at 5 universities currently uses an expensive commercial EDC that costs $40K/year and is inflexible. They want to migrate to something more affordable and customizable.

Their approach:

```
# IT staff deploys to AWS
./aws_setup.sh \
  --region us-west-2 \
  --study-name "Multisite ADHD Trial" \
  --study-id "ADHD-MULTI-2025" \
  --admin-password "SecurePassword123!" \
  --domain adhd-trial.org \
  --instance-type t3.large  # Larger instance for multi-site
```

Result: - Single centralized instance accessible from all 5 sites - HTTPS with automatic security certificates - Role-based access: 2 administrators, 5 principal investigators, 20 coordinators, 5 data managers - Data shared securely across institutions - Monthly cost: $40 (95% cheaper than commercial EDC)

Cost: $40/month infrastructure + staff time for administration **Timeline**: 1 week from decision to enrollment opened

---

5.3 Scenario 3: Individual Investigator Independence

Dr. Robert was running a study with a local Biostatistics Lab that managed his EDC. The lab relationship deteriorated due to cost escalation and support issues. He wants to take control.

The situation: - His data is in a commercial EDC (vendor owns the data format) - Migration would cost $20K to export and reformat - He needs a system that's independent from any vendor - He has basic IT skills but doesn't want to manage Linux servers

His solution: 1. **Deploy ZZedc independently** on AWS (not through the Biostatistics Lab) `bash ./aws_setup.sh --region us-west-1 --study-name "Robert's Study" \ --study-id "ROBERT-2025" --domain robert-study.org`

2. **Migrate his data** from the old system to ZZedc
    - Old system exports to CSV
    - CSV imported into ZZedc
    - Data is now in standard SQLite format
3. **Complete his study independently**
    - Data entry continues in ZZedc
    - Analysis done with R/Python (direct database access)
    - Final data archived as standard database file
4. **End of study**
    - Data archived in standard SQLite format to institutional repository
    - System shut down (delete EC2 instance)
    - Ongoing data access requires only free SQLite tools

Result: Dr. Robert owns his data and system. If he wants to work with another Biostatistics Lab in the future, he can—without vendor lock-in.

---

6 Getting Started: Step-by-Step

6.1 For Solo Researchers

Requirements: R installed, ~5 minutes

```r
# Step 1: Install
install.packages("zzedc")

# Step 2: Load and initialize
library(zzedc)
zzedc::init()

# Step 3: Answer interactive questions
# The system guides you through setup

# Step 4: Launch
```

```
Sys.setenv(ZZEDC_SALT = "...")  # (from setup output)
launch_zzedc()

# Step 5: Open browser to http://localhost:3838
# Step 6: Login with admin credentials you chose
```

Detailed instructions: See `vignettes/quick-start-solo-researcher.Rmd`

6.2 For AWS Deployment

Requirements: AWS account, AWS CLI, ~15 minutes plus setup time

Step 1: Prepare

```
# Ensure AWS credentials configured
aws sts get-caller-identity

# Gather information
# - Study name
# - Study protocol ID
# - Domain name (e.g., trial.example.org)
# - Admin password (8+ characters)
```

Step 2: Deploy

```
cd deployment/
chmod +x aws_setup.sh

./aws_setup.sh \
  --region us-west-2 \
  --study-name "Your Study" \
  --study-id "YOUR-STUDY-ID" \
  --admin-password "SecurePassword123!" \
  --domain trial.example.org \
  --instance-type t3.medium
```

Step 3: Point domain to instance - Wait for DNS to propagate (can take up to 24 hours) - Access application at `https://trial.example.org`

Detailed instructions: See `deployment/AWS_DEPLOYMENT_GUIDE.md` or `vignettes/quick-start-aws-devops.Rmd`

6.3 For Migration from Other Systems

Migrating from commercial EDC, Excel, or other sources?

1. **Export from source system** (typically CSV or Excel)
2. **Import into ZZedc** using the data loader tools
3. **Verify data quality** in ZZedc's validation interface
4. **Complete study** in ZZedc
5. **Archive final data** as standard SQLite database

Detailed migration guides available in documentation.

---

7 Long-Term Sustainability

7.1 Ownership and Control

When you deploy ZZedc, you own: - **The infrastructure**: Your EC2 instance, your VPC, your data storage - **The data**: Standard SQLite format, completely portable - **The system configuration**: You control every setting - **The codebase**: Open source on GitHub; you can fork and modify if needed

7.2 Maintenance and Updates

Monthly: - Monitor disk usage and instance health - Check logs for errors

Quarterly: - Update OS packages on EC2 - Update ZZedc package if new version available - Test backup/restore procedure

Annually: - Security audit - Capacity planning (do you need a larger instance?) - Archive completed studies

See `IT_STAFF_DEPLOYMENT_CHECKLIST.md` and `IT_STAFF_TROUBLESHOOTING.md` for detailed guidance.

7.3 Migration Path

If you want to migrate away from ZZedc in the future:

1. **Export database**:
   ```
   # SQLite is a standard database format
   sqlite3 zzedc.db ".dump" > database_export.sql
   ```
2. **Access with any tool**: Your data can be accessed by any SQL tool, Python, R, Stata, SAS, etc.
3. **Complete ownership**: No vendor locks, no proprietary formats

---

8 Comparison: ZZedc vs Commercial EDC

Feature	ZZedc	REDCap	Medidata
Cost	~$40-500/month infrastructure	$200-1000/month licensing	$500-2000/month licensing
Data ownership	Complete (standard SQLite)	Vendor owns servers	Vendor owns servers
Customization	Full source code access	Limited data dictionary	Limited, expensive
Long-term sustainability	You control it	Dependent on vendor	Dependent on vendor
Compliance frameworks	GDPR, CFR Part 11 built-in	Available as add-on	Available as add-on
Vendor lock-in	None (open source)	Moderate (proprietary formats)	High (proprietary system)
Support model	Community + self-support	Commercial support	Commercial support
Learning curve	Moderate (R/Docker knowledge helpful)	Low (point-and-click)	Low (point-and-click)

---

9 Conclusion: Taking Control

Clinical research data management doesn't require expensive, proprietary vendors. ZZedc demonstrates that open-source tools can provide **security, compliance, and ease-of-use** while maintaining your independence.

Whether you're: - A solo investigator seeking a simple, free EDC for a pilot study - A research team tired of vendor pricing and inflexibility - An established researcher wanting to migrate away from vendor lock-in - A multi-site consortium seeking affordable, scalable data management

ZZedc offers: - Complete investigator ownership and control - Low or no licensing costs - Standard, portable data formats - Security and compliance by default - Flexible deployment (local, cloud, hybrid) - Open source (fork and customize as needed)

9.1 Next Steps

Start exploring: - **Solo researcher?** See `vignettes/quick-start-solo-researcher.Rmd` - **Team with AWS?** See `vignettes/quick-start-aws-devops.Rmd` - **IT staff deploying?** See `deployment/IT_STAFF_DEPLOYMENT_CHECKLIST.md` - **Need help?** Visit https://github.com/rgt47/zzedc for documentation and issues

The goal of ZZedc is simple: **Put clinical research data management back in the hands of investigators**.

9.2 Resources

- **ZZedc GitHub**: https://github.com/rgt47/zzedc
- **Documentation**: See `vignettes/` directory for comprehensive guides
- **Deployment guides**: See `deployment/` directory for AWS, Docker, and operational guides
- **Support**: Open an issue on GitHub or contact zzedc@ucsd.edu

---

Published: December 2025 **Author**: Clinical Research Technology Team **License**: Open source (see GitHub repository for license details) **Status**: Production ready