A Journey of Discovery With SCAD Software

By Robert Schifreen

SCAD Software invited IT journalist, writer and trainer, Robert Schifreen, to meet with their founder, Andy Fensham and International Managing Director Shaun Orpen, to explore the "SCAD Methodology". It's fair to say he was surprised by what he learned. This is his story in his own words

Despite extensive IT experience and having read through the SCAD Software web site and presentations, I still didn't "get" what SCAD Software really did. Was I being sold a service or a physical product? Is it a framework or a methodology, or an outsourcing service? What has metadata got to do with it? Is it aimed at corporates or startups? Why would I want to reengineer a complex application which is working just fine, purely to reduce the number of lines of code?

There was clearly an "Aha! moment" in there somewhere, but I couldn't see it. Having spoken with Shaun and Andy, everything began to make sense. So, let me try to explain in my own words what SCAD is, and the main USP, and the sort of company that might want to consider using it.

SCAD Software: The USP

Over time, business applications become more and more critical to the ongoing success of a company. New features are continually added, as a result of product enhancements or regulatory changes.

While the underlying database might be robust, the methods through which customers and staff interact with it become ever more complicated. The applications also become harder to manage and maintain.

In the financial services industry for example, it is not uncommon for a key application to comprise hundreds of database tables, thousands of data entry screens and views, and a million lines of code. Crucially, making even a minor change to the system is time-consuming and prone to errors, often requiring weeks of development and testing. If a new field or column is added to the customer database, every part of the application which references that new field needs to be updated. This could be thousands of separate screens, views and reports.

Similarly, if a change in audit or compliance regulations means that employees below a certain level must not be given write privileges on a particular database field, this commonly requires thousands of changes to the core application across hundreds of separate files to ensure that the new directive is applied across every screen.

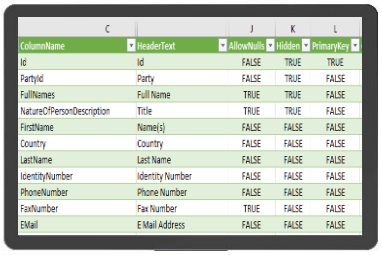
If the application is launched into a new territory in which a different language is spoken, all of those thousands of files that make up the screen and view definitions need to be duplicated, with just the textual content altered. Thus, instantly doubling the number of lines of code that now need to be maintained.

SCAD Software changes all this. They re-engineer your application to drastically reduce the amount of source code, typically by at least 40%. Then they host and fully manage it for you on your preferred cloud provider's infrastructure. Your developers and DBAs are freed up to spend their time improving and innovating for your business and your customers, instead of being involved in unending code maintenance.

The Proprietary Metadata-Based Methodology

The key to SCAD's re-engineering process and drastic source code reduction is their proprietary metadata-based methodology.

Metadata is added to the database alongside existing fields/columns. This defines how a field is to be displayed on a screen, who is permitted to see it, whether they can also edit it, and so on.



ColumnName	HeaderText	AllowNulls	Hidden	PrimaryKey
Id	Id	FALSE	TRUE	TRUE
PartyId	Party	FALSE	TRUE	FALSE
FullNames	Full Name	TRUE	TRUE	FALSE
NatureOfPersonDescription	Title	TRUE	TRUE	FALSE
FirstName	Name(s)	FALSE	FALSE	FALSE
Country	Country	FALSE	FALSE	FALSE
LastName	Last Name	FALSE	FALSE	FALSE
IdentityNumber	Identity Number	FALSE	FALSE	FALSE
PhoneNumber	Phone Number	FALSE	FALSE	FALSE
FaxNumber	Fax Number	TRUE	FALSE	FALSE
EMail	E Mail Address	FALSE	FALSE	FALSE

Figure 1 Metadata Presented in an Excel Table

Remember those thousands of data entries and editing screens, comprising tens of thousands of lines of code?

They're all redundant. Screens are now built dynamically on the fly, in a browser or a progressive web app, based on the information contained in the metadata.

You can even interact with the application via its built-in REST API, to provide interoperability with other new or legacy systems.

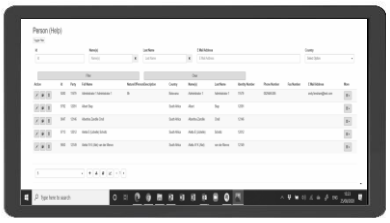


Figure 2 The Completed screen rendered automatically from the metadata

An endpoint for each screen is created automatically and consistently with no additional development work required, and these can be called from anywhere via any programming language. All major authentication methods are supported.

Making a minor change to the application, such as adding a new field to a customer record or changing the permission level required to edit it, requires nothing more than a change to the metadata that can be done in minutes.

All the screens will then incorporate the change automatically. No source code changes are required.

And because the application is fully managed by SCAD, you simply need to request the change via the correct channels and it will be implemented. Initially on your development or UAT environment. When you're happy with it, sign it off and it goes live instantly. If there's a problem you can roll back just as quickly.

When you launch your system in a new territory, create a new supplementary set of metadata that contains just the lines of text in the new language. These will override the text from the original version but everything else stays the same. No duplication of screen or view definitions is required.

Your new system is always fully documented, because the metadata IS the documentation.

- No longer will there be anomalies between what your outdated documentation says should happen in theory and what actually occurs in practice.
- No longer will missing system documentation be a problem, because you can simply look at the metadata to learn exactly how a particular feature will behave.
- And by using a report generator you can extract that metadata and create documentation in any format that you like, which is guaranteed to be accurate. If the metadata specifies that only staff in Marketing can change a customer's email preferences, the documentation will say that too, and the code is guaranteed to behave precisely that way every time, across every screen

Proven In Practice

SCAD has extensive experience with customers in markets such as banking and finance, CRM, insurance, manufacturing and distribution. The SCAD method is also ideal for startups who are building an MVP. It's incredibly quick to create a proof-of-concept application which can quickly and easily be revised in response to user testing and feedback. Just change the metadata, not the code.

Working with the team at SCAD to build new applications, or port existing ones, to the metadata methodology is incredibly quick. In one case study, a major financial institution built a system for managing and approving billions of dollars in loans. After a comprehensive exercise to capture requirements the system was built, tested, and rolled out to 1,000 branches, in just 3 months, saving an estimated $20m compared to doing things "the old way".

Of course there's more to a complex business system than the database and the screens. Behind it all is the business logic which governs the content of those screens and views. Whether the loan being applied for has been approved, for example. Or whether the order should be marked as having been fulfilled. Where the existing logic is implemented using modern best-practices such as stored procedures, the SCAD team will re-engineer it, incorporate all necessary metadata, and build the same functionality into the newly designed low-code system. Where the business logic is not so accessible, such as when it's contained in conventional source code, SCAD will normally leave it in-situ and call it directly from the new metadata-based system in order to display the necessary screens and views. Over time, and once the new system is up and running, this legacy business logic can then be refactored and modernised and gradually brought into the new application

In Summary

The team at SCAD make it easy for large, complex legacy systems to be transformed into modern, agile applications with a vastly reduced codebase, allowing expensive resources such as on-premises servers as well as DBAs and developers to be freed up. In addition, when systems need to be rewritten in a new programming language, such as to take advantage of new best practices or to migrate away from free open-source languages that have become proprietary and expensive, SCAD can get the job done quickly.

- - oOo --

About Robert Schifreen

Robert Schifreen is a writer, trainer and commentator on IT and software security. Until recently he was engaged in the IT department at the University of Brighton, latterly as a SharePoint farm administrator, where he architected, built and maintained a system with 5,000 daily users across 14 servers. He is author of the book "Defeating the Hacker - A Non-Technical Guide To IT Security" and was editor of the UK magazine for software developers, .EXE Magazine, and Software Management magazine for software managers.

Put us to the test - ask about our Business Accelerator Programme

If you have any questions, we are happy to discuss the advantages of using SCAD Software.
shaun.orpen@scadsoftware.com **+447900 681725**