



Paper TT09

Back to the Studio: Why SAS Studio® Merges the Past, Present and Future of Programming Clients

Paul Shannon, Amadeus Software, Witney, UK

ABSTRACT

Many SAS users are programmers first, despite the many graphical interfaces available for SAS®. Users in the Pharmaceutical industry rely heavily on SAS Enterprise Guide and SAS Display Manager for their default programming client. SAS Studio is a relatively new browser-based alternative that provides familiarities for users of both Enterprise Guide and Display Manager, whilst offering powerful new enhancements such as extensibility, Git integration for source control as well as hosting and administration benefits for IT support teams.

This paper demonstrates some of the familiarities with existing solutions, demonstrates new enhancements and discusses important benefits for migration, such as simplicity of access across multiple devices, simpler hosting and maintenance considerations for platform engineers and cloud or on-premise support for any individual or company considering adopting SAS Studio as a programming environment.

INTRODUCTION

SAS users in the Pharmaceutical sector have traditionally favoured the use of programming environments in SAS. Traditionally, this was via SAS Display Manager (sometimes referred to as Base SAS). Twenty years ago, SAS first released Enterprise Guide, alongside SAS v8. Suddenly SAS didn't have to be installed on your computer, merely a thin client. Whilst Display Manager is a programming interface first, Enterprise Guide offered a new user experience, with many point-and-click tasks to automate programming steps, intellisense and programming tips to name a few.

The last twenty years has seen significant change in computing power and the way users interact with computers and software has progressed immensely. Similarly, the SAS client applications have changed too. Products such as SAS Visual Analytics have provided data scientists the tools to analyse and share reports without the need for programming knowledge, thus increasing the potential software user base. That said, the role of a SAS programmer is as important as ever. But with smartphones, tablets and laptops being a common tool for many users, it is more important than ever to have a common programming environment that offers a consistent interface for the programmer. For IT support providers, the exponential increase in devices and types means installing software on devices in the traditional, controlled manner seen in many large organisations would pose a logistical nightmare.

SAS Studio is a programming application delivered to the end user via their browser. This offers a single point of control for IT administrators and platform engineers, whilst offering security controls to satisfy the most stringent of regulations. As cloud computing becomes an increasingly frequent presence in organisations of all sizes, the latest incarnation of the SAS platform, SAS Viya is designed to support cloud or on-premise deployments with similar ease. This further increases the demand for a single-point of access to SAS clients, compatible across different types of computing device.

This paper addresses the issues above to demonstrate how the adoption of SAS Studio offers a familiar, yet modern programming environment whilst offering the most secure and cost-effective method for deployment and platform administration.

BACKGROUND TO SAS CLIENT APPLICATIONS

You may be asking the question, “what is SAS Display Manager”? Others may ask the question “What is SAS Enterprise Guide?”. Some may be more familiar with these names than others, some may consider these simply as “SAS”, with the names interchangeable across these functions. Let us try to explain the differences concisely.

SAS Display Manager is the default program window used for the compilation parsing and execution of programs when SAS is installed directly on your computer. You may also refer to this as “classic” SAS. We should be wary of such a term as the word “classic” is somewhat subjective. The current version of SAS Display Manager is shown in the screenshot below.

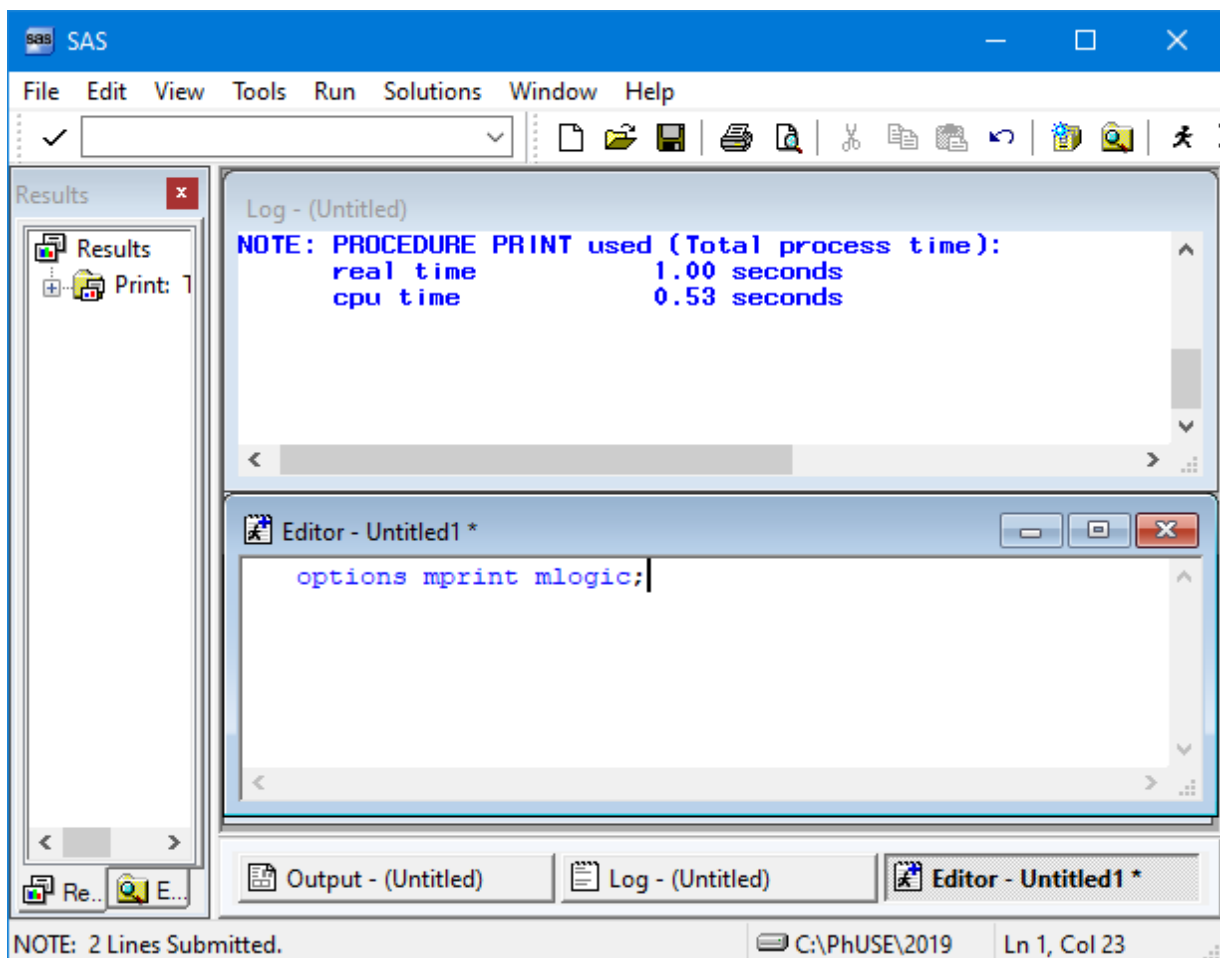


Figure 1 - SAS Display Manager

Enterprise Guide is a thin-client, used to connect to one or more remote SAS servers, without the need for installing the resource heavy SAS software directly on your computer. This is the more modern interface introduced alongside the SAS Intelligence Platform. Enterprise Guide provides the option of a more point-and-click interface, complete with the ability to write code. In-fact many SAS users predominantly write code and forego the point-and-click functions, such that updates in the latest release of Enterprise Guide reflects this. Those readers who are familiar with SAS Enterprise Guide will be familiar with the figure below:

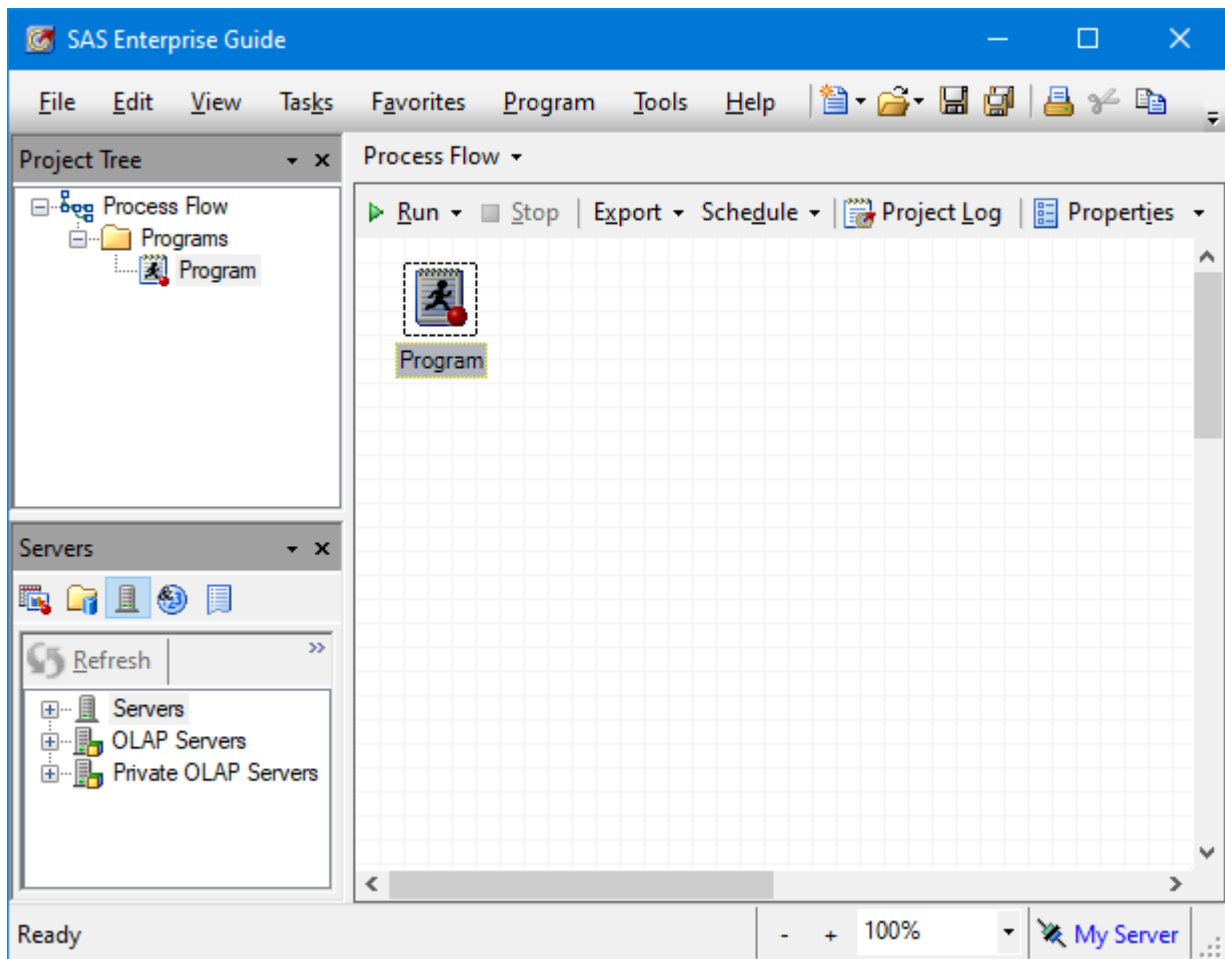


Figure 2 - SAS Enterprise Guide (v7.15)

Now we understand what Display Manager and Enterprise Guide are, it's time to answer the question, "What is SAS Studio?" Is this just another change for the sake of change? Is it another tool to do the same job? Is it worth the investment deploying and re-training staff, without offering any major enhancements other than a slightly more polished appearance?

Firstly, SAS Studio is a browser-based thin-client. Much like SAS Enterprise Guide, SAS Studio is designed to connect to a remote server running the SAS platform; there is no need to have SAS installed directly on each users' computer. The term "browser-based", means exactly that – it is a web application used via your usual web browser, e.g.: Google Chrome, Microsoft Edge, Firefox and Apple Safari.

Is SAS Studio a change for change's sake? The way we use and interact with computers continually evolves. When this author started as a fresh-faced intern in the pharmaceutical industry, having a laptop was the preserve of senior managers only. Today we think nothing of browsing the internet and shopping on our portable devices. By having a browser-based editor, SAS programming can finally become mobile, on your tablet or phone, should the need arise. Notwithstanding the logistics of typing code on a touchscreen keyboard. The centralisation of the software makes the deployment and maintenance far simpler than previously. Thus, it's safe to say SAS Studio offers real-world benefits to both the programmer and platform administrator.

Most importantly of all, SAS Studio has features loved by programmers from both SAS Display Manager, and Enterprise Guide.



SAS STUDIO AND THE PAST – INCORPORATING THE BEST OF SAS DISPLAY MANAGER

THE USER INTERFACE

Display Manager users will be familiar with two key areas of the window when using SAS; the explorer window (to the left) and the main code area, showing code, log and results windows. Figure 3 shows an example layout of SAS Studio.

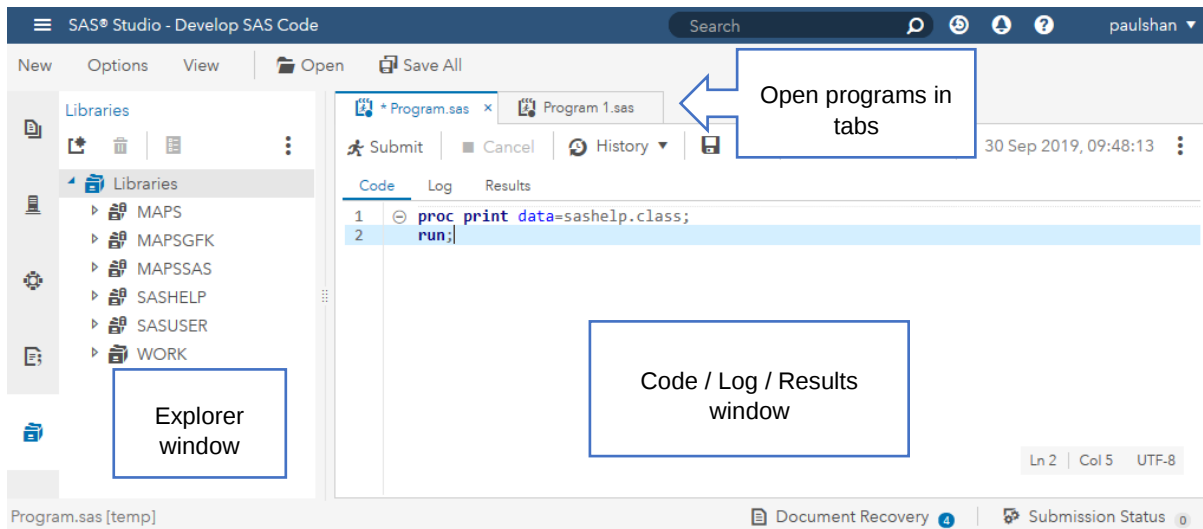


Figure 3 - SAS Studio

THE SYNTAX EDITOR AND CUSTOMISING COLOURS

Many programmers have colour schemes and preferences when programming in SAS. Some may remember a time before syntax editors, which was the dark (monochrome) program editor. SAS Studio carries over the enhanced editor customisation options, enabling individuals to configure their preferred colour scheme and font size. This is far from a “nice-to-have” feature, the Web Content Accessibility Guidelines (WCAG) 2 recommends many features to ensure all users can read and distinguish a site or application. Specifically, a user with colour-blindness can set a colour combination that is distinguishable. Enabling bold and italics for certain keywords means that colour is not the only visual means of identifying keywords. Similarly, font size used is important to users with less than 20/20 vision. Whilst this may seem to some as basic functionality, it's imperative to ensure all SAS programmers can work effectively.

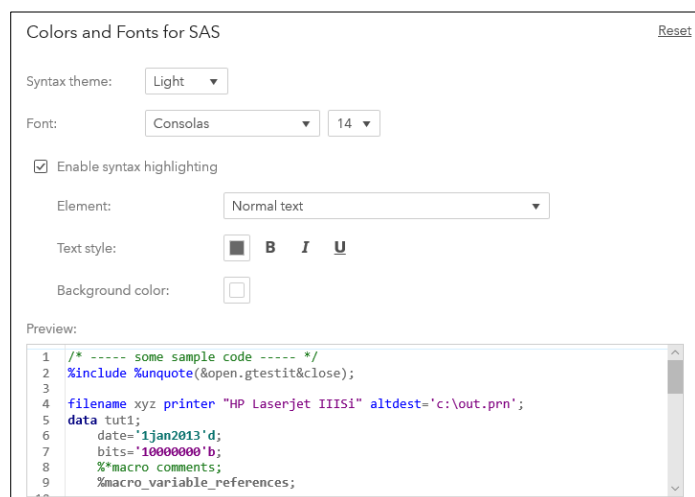


Figure 4 - Colour and Font Settings in SAS Studio



TEXT ABBREVIATIONS / CODE SNIPPETS

SAS Display Manager has long contained a tool for defining text or code snippets (Abbreviations). SAS Studio has a similar function, but goes by the name of “Code Snippets”. In Display Manager, users were given a simple dialog with a plain text editor only for creating an abbreviation.

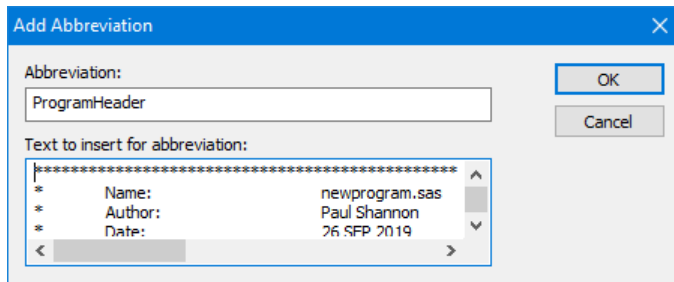


Figure 5 - Setting Abbreviations in Display Manager

Calling or inserting an abbreviation wasn't straight-forward, the user had to know the name given to the abbreviation. The preview was limited, and the name had to be typed entirely accurately, including case. This author has felt frustration on many occasions with these limitations in years past.



Figure 6 - Using Abbreviations in Display Manager

With code snippets in SAS Studio, significant improvements have been made. Firstly, creating snippets are much easier, by simply opening a new program window. This provides syntax editing in your snippet. The “Add as Snippet” toolbar button is used to save the current program as a snippet.

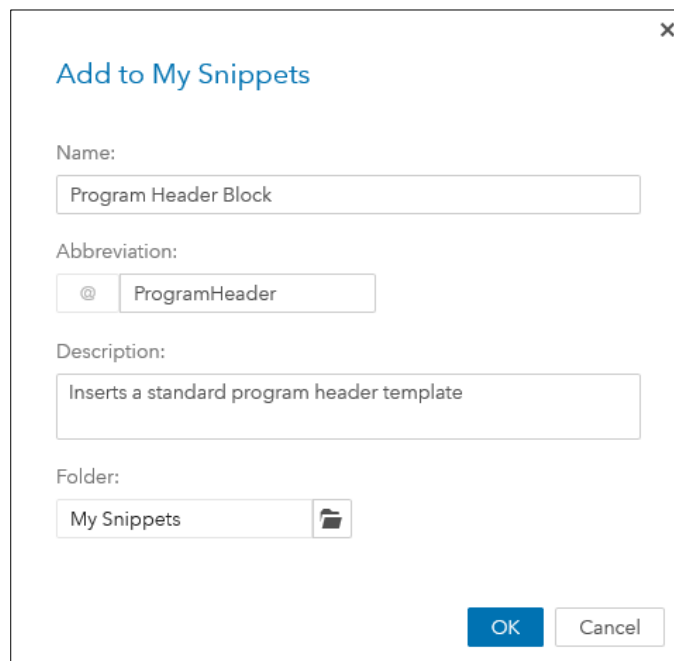


Figure 7 - Creating a Snippet in SAS Studio



After completing this step, you will be able to see the snippet in My Snippets (or the folder you select). Snippets are found in the explorer window on the left side

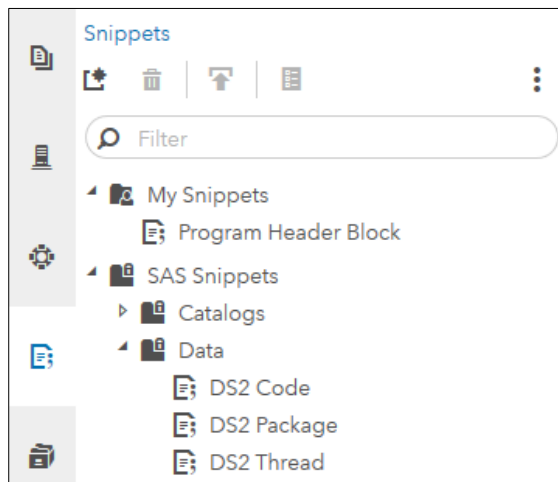


Figure 8 - My Snippets

You can add this or any snippet to your code by drag-and-dropping the snippet on to your program. If you specified an Abbreviation when configuring your snippet, you can alternatively insert a code snippet by typing the abbreviation with an @ prefix with the abbreviation. In the example above, typing @ProgramHeader, followed by TAB or ENTER, will insert the relevant snippet. The next screenshot demonstrates the filter and preview function provided when you start typing an abbreviation using the @ prefix.

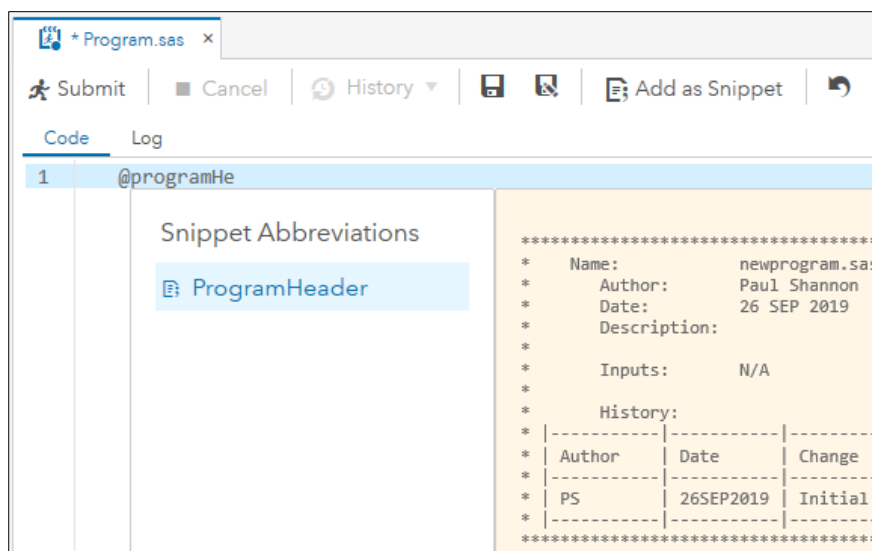


Figure 9 - Using a Snippet in SAS Studio

It is possible to share snippets with other users; a practice that is particularly useful for ensuring a best-practice and consistent programming methodology across a team, or even multiple teams of programmers. However, there is no specific sharing function. Instead, the simplest option is to add your snippet(s) to the Explorer window. This can be performed using the right-click menu from the snippets window, and choosing to save the snippet to the Public SAS folder, or another suitable location.

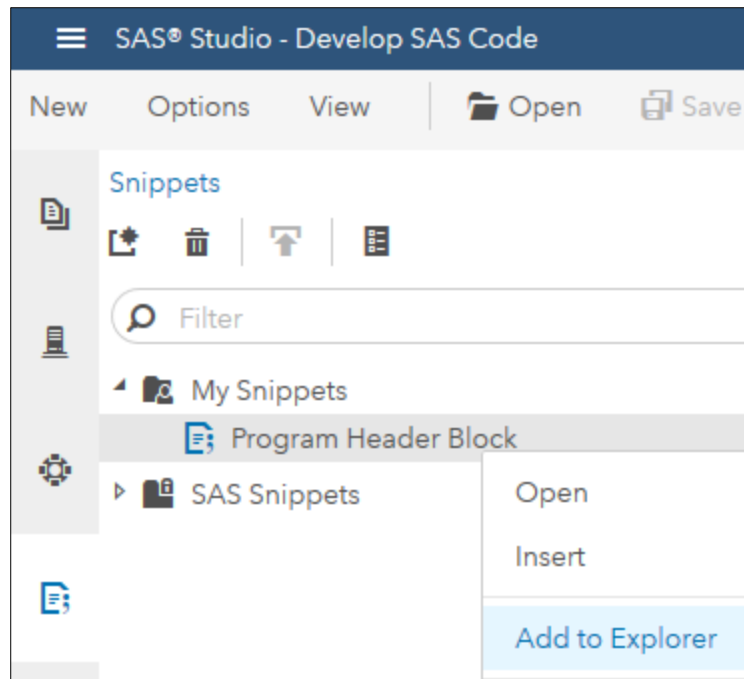


Figure 10 - Sharing a Code Snippet, by Adding to Explorer

An alternative (and arguably better) method would be to store the source files for snippets in a Git or NPM repository. This will allow any user to pull necessary repositories to a local instance and use within their SAS Studio instance. This also allows for clearer management of snippets over the lifecycle of code snippets.

SAS STUDIO AND THE PRESENT – INCORPORATING THE BEST OF SAS ENTERPRISE GUIDE

SAS Enterprise Guide was first released in 1999, to coincide with SAS v8 and the inaugural release of SAS Metadata Server. Not only did this provide the option for thin-client/server architecture, but several point-and-click options were introduced, such as Tasks, Wizards and the Process Flow. Happily, these functions are broadly available in SAS Studio (with some caveats, explained later). A quick review of the SAS Studio window in the figure below highlights the items familiar to Enterprise Guide users.

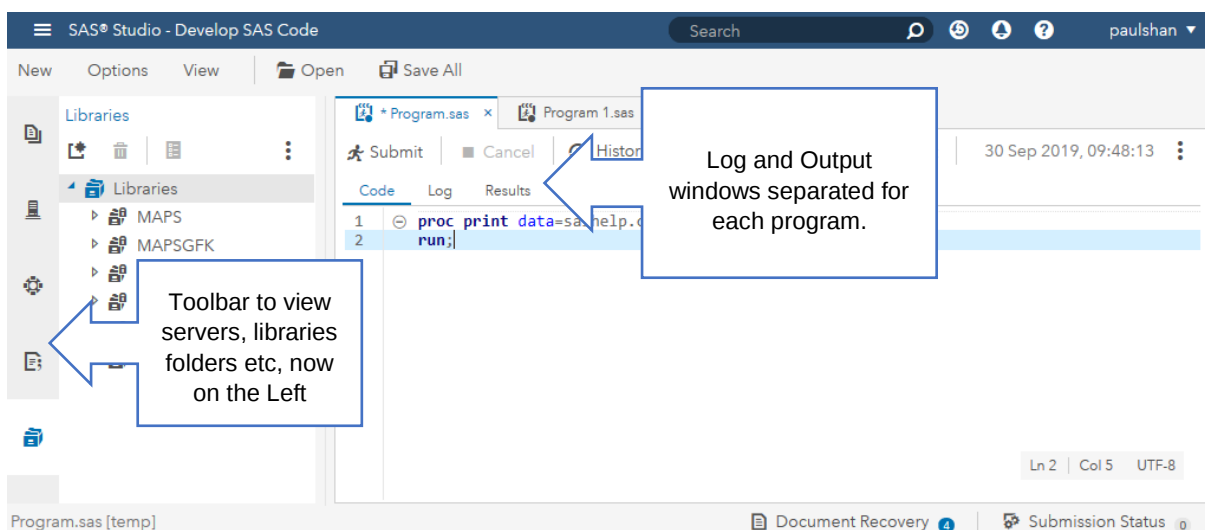


Figure 11 - SAS Studio - Similarities with Enterprise Guide



INTELLISENSE PROGRAM WRITING

SAS Enterprise Guide brought intellisense program editing to SAS, as is often seen in other development environments, whereby the programmer is guided towards correct syntax by short pop-up windows. In the example below, the syntax editor in Enterprise Guide recognises the APPEND procedure is referenced, thus shows options available for this procedure as the user continues to type.

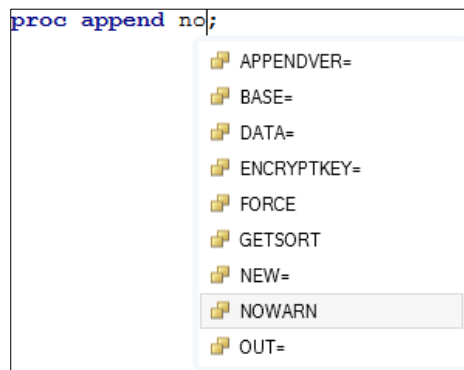


Figure 12 - Intellisense Menu in Enterprise Guide

Help text is available for code written in the syntax editor by simply hovering over a keyword. In such cases a large tooltip appears with help text on the item being selected and provides immediate links to SAS product documentation, code samples and papers.

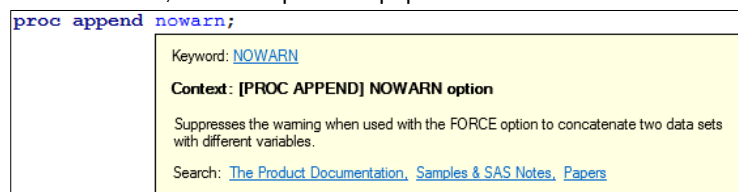


Figure 13 - Help Tooltip in Enterprise Guide

In SAS Studio, the functionality and user-interface of this feature has improved. Detailed help information now shows alongside the intellisense menu, rather than having to complete the syntax before coming available to the end user.

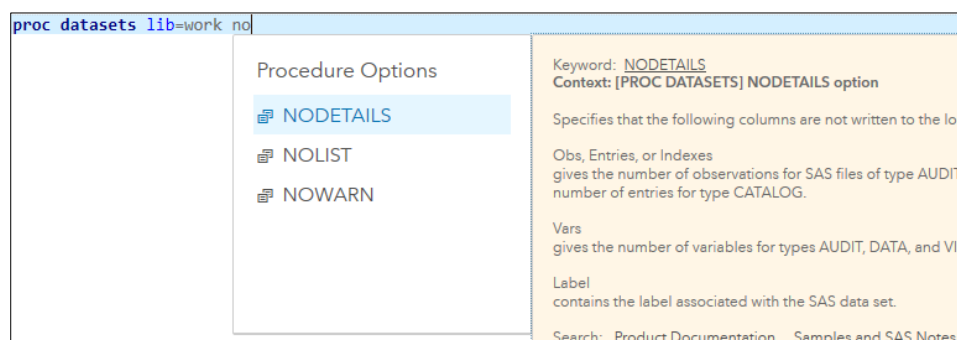


Figure 14 – Intellisense and Tooltips in SAS Studio

POINT-AND-CLICK PROJECTS – TASKS

Pre-defined tasks are available in SAS Studio in much the same way as SAS Enterprise Guide, although they are located slightly differently in the user interface. There is no-longer a “tasks” menu, instead you will find tasks in the vertical toolbar. Some additional tasks now exist which align to the advantages of SAS Viya, which may not



be present for users based on SAS 9.4. Subject to the modules in your SAS license, the traditional tasks are present, grouped by categories such as data manipulation, statistical analysis, which have changed a little from Enterprise Guide.

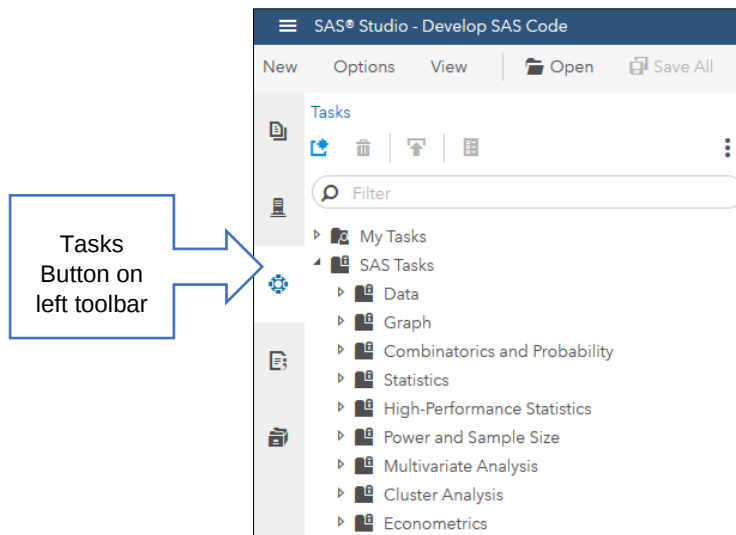


Figure 15 - Available Tasks in SAS Studio

Furthermore, developing your own tasks is now far simpler. With SAS Enterprise Guide, development of custom tasks could be achieved by developing an add-in using Microsoft .NET. Papers have previously been written on such topics, and this paper does not intend to cover such detail. The problem is that, for a SAS programmer to create an add-in, they must also have knowledge of Microsoft .NET's programming languages such as C# or VB. They must also have a license for Visual Studio. This is before we consider the cost of deployment and maintenance of bespoke add-ins.

SAS Studio takes the development of tasks in-house. Tasks are built directly within SAS Studio and created using the XML language. SAS includes a user guide for developing custom tasks in SAS Studio. Development is made easier by enabling source code from pre-defined tasks to be copied and adapted. Programmers are therefore able to quickly upskill on the required XML syntax, and the code options available.

SOURCE CODE CONTROLS

SAS Enterprise Guide provides a version control solution, for managing changes to SAS programs. This was limited in earlier versions of Enterprise Guide to program code embedded within a project file, with no support for external program files. More recent versions of Enterprise Guide have compatibility for integration with Git repositories, but requires third-party management to be truly effective and follow best practice DevOps processes. The use of SAS Studio throws up a new issue; as a browser application, your SAS programs must be saved on the remote server, browsers have no programmable access to your local file system (for obvious security reasons).

There are Git functions within the base SAS language to handle most of the steps used to manage source control. At this stage, you may be thinking this is a step backwards, why must one write code to manage code? Admittedly, this does seem somewhat of a misnomer to progression. Nevertheless, genuine Git integration within SAS has not been available until recently. We will discuss how SAS Studio can be used to improve the DevOps process later.

THE PROCESS FLOW

The Process Flow is available in SAS Studio v3.x. This closely follows the corresponding feature in Enterprise Guide. The principal is to offer a node-based flow to your SAS projects, enabling a visual programming environment, over simple plain coding. Adding tasks, programs and setting a sequence for execution is much the



same as in SAS Enterprise Guide. One visual difference is the use of display nodes, rather than icons. This design is in keeping with other SAS products, such as Data Integration (DI) Studio.

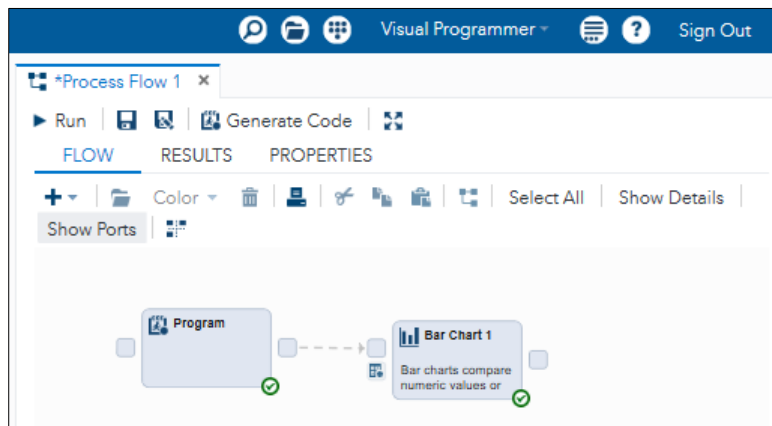


Figure 16 - Visual Programmer Mode in SAS Studio 3.8

SAS STUDIO & THE FUTURE

So far, we have seen several examples where SAS Studio takes features from SAS Display Manager and SAS Enterprise Guide. This section explores how SAS Studio benefits from further development. Simply re-creating the same application in a web browser can't be considered the next-gen client. Indeed, it is not.

COMPATIBILITY: CROSS PLATFORM, CROSS DEVICE,

SAS Studio (as with most other SAS user interfaces), is supported on all the main web browsers:

- Apple Safari 10.0 and later
- Google Chrome 61.0 and later
- Microsoft Edge 40.1 and later
- Mozilla Firefox 56.0 and later

Formal support for mobile devices is not yet available; although there's nothing to physically stop a user logging in via a mobile device, even if the user interface is not optimised for such screen sizes, particularly phones. Nevertheless, this still opens SAS to a much simpler mechanism for deployment to end users. Consider even today, there will undoubtedly be presenters who require the use of their own laptop, to demonstrate with specific client software. Browser applications do away with the need to deploy specific software to each client endpoint, instead centralising the maintenance and updates to the SAS server only.

ADMINISTRATION: ONE INSTALLATION, MULTIPLE USERS

Many SAS programmers may not give too much consideration to the deployment of SAS on a server, there are indeed considerable differences in skillsets between administrators and analysts. SAS is usually deployed on one or more servers, separating the architectural layers of SAS software. The SAS Middle Tier, or "Mid-Tier", is used to host web applications, such as SAS Studio, Visual Analytics and others (depending on your company's licensed software). A single deployment of SAS Studio on the Middle Tier is used to serve all users with the SAS Studio browser application.

Having a single point of deployment means SAS is less burdensome on desktop support teams. Removing the need for deployment of software to every individual client computer has a time-saving benefit, that is broadly proportional to the number of users. When preparing to deploy SAS 9.4, the diagram below shows a common example of the software to be installed on each server, on a 3-server architecture. Client software is deployed to each user's PC, thus the effort is a multiple of the users.

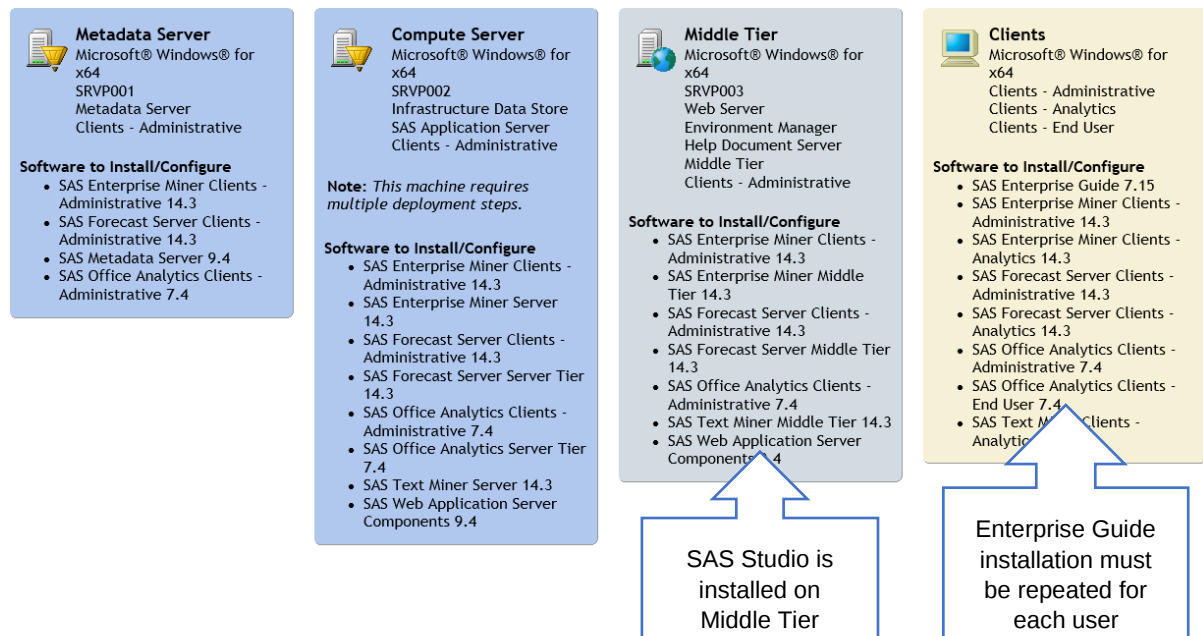


Figure 17 - SAS 9.4 Installation Summary Diagram

Advantages of consistency and release management are also provided by browser applications. Imagine the scenario when SAS (or any software vendor, for that matter), release an update or upgrade to their client applications. Applying this update/upgrade to SAS Studio involves applying the update to the relevant server, once. All users are (by definition) updated the next time they log in.

Conversely, many users of Enterprise Guide will, be familiar with the prompt below, informing of updates available. When the end-user is prompted to install updates, inconsistencies are inevitable within a given organisation between the exact version of Enterprise Guide in use. Furthermore, some users will not have the administrative privileges required to install updates, meaning prompts for updates are merely a distraction.

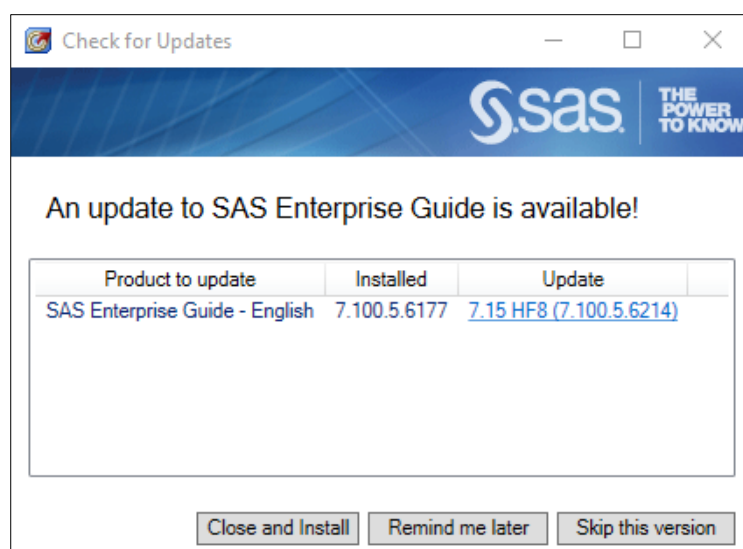


Figure 18 - SAS Enterprise Guide "Check for Updates" Dialog



End-user updates may also be dependent upon a user with elevated privileges, which returns to the original issue of requiring IT resource to deploy the software update in each client PC. In summary, SAS Studio offers a faster, simpler deployment and release management framework, allowing future updates to be rolled out to an organisation centrally as well as ensuring consistency across the company.

HOSTING SAS IN THE CLOUD

For those who are not familiar with the finer details of cloud hosting, some describe the cloud in its simplest form as “using someone else’s server”. Cloud computing opens a host of benefits for software with supporting architecture, for which SAS Viya has been designed and built in mind. SAS Studio is deployed in much the same method whether SAS is being hosted on-premise or in the cloud. SAS Viya allows applications to make use of “elastic computing” models, whereby the performance and capacity of hosting servers can increase and decrease according to demand, within minutes. This has the potential to improve the performance at busy times, without the SAS programmer being aware of any changes being made.

ENHANCING THE SAS PROGRAMMING TEAM WITH DEVOPS PROCESSES

DevOps is a set of practices that automates the processes between software development and IT teams, in order that they can build, test, and release software faster and more reliably. One of the key principles is to have a source control repository for storing, versioning and managing promotion/release of programs.

The term “source control”, differs from “version control”; version control is the process of tracking versions of programs. SAS programmers will often apply this by versioning files as saved, with comments in a program header. Source control goes much further, whilst managing versions of a given program or file, features such as branching and merging means that multiple programmers can work on the same program suite simultaneously without risk of conflicting or overwriting each-others work. Git is a source control solution that is widely used throughout the software industry. This paper is not an introduction to, or a technical exercise in Git or any other source control solution, so there is little focus on the precise functionality. However, below is a simple concept of how to make SAS Studio work as seamlessly as possible.

Consider the simple data step below. This demonstrates the use of the `gitfn_clone` function, which clones a remote repository for editing.

```
data _null_;
  rc = gitfn_clone(
    "https://github.com/sascommunities/sas-dummy-blog/",
    "/home/paulshan/repos/sas_dummy_blog"
  );
run;
```

A total of 24 `GITFN_` functions are in base SAS, and the topic of managing source code via a Git repository is worthy of a separate paper. However, the code examples below demonstrate examples of performing the most common Git commands; Pull, Commit and Push.

```
data _null_;
  rc = gitfn_pull(
    "/home/paulshan/repos/sas_dummy_blog",
    "&sysuserid.",
    "{SAS002}F418233159F49DEB5770F40F17936BA5276E5DFB0CAD2FD3"
  );
run;

data _null_;
  rc = gitfn_Commit(
    "/home/paulshan/repos/sas_dummy_blog",
```



```

    "HEAD",
    "&sysuserid.",
    "&sysuserid.@company.com",
    "Description of the change(s) made in the source code"
);
run;

data _null_;
  rc = gitfn_push(
    "/home/paulshan/repos/sas_dummy_blog",
    "&sysuserid.",
    "{SAS002}F418233159F49DEB5770F40F17936BA5276E5DFB0CAD2FD3"
  );
run;

```

One option is to create custom tasks in SAS Studio which can simplify the process of performing routine source control tasks. Now we no-longer need the assistance of a Microsoft .NET developer, or help with deploying and installing add-ins, it's possible to create and deploy custom tasks quickly and easily. Below is an example of creating a custom task to clone a Git repository.

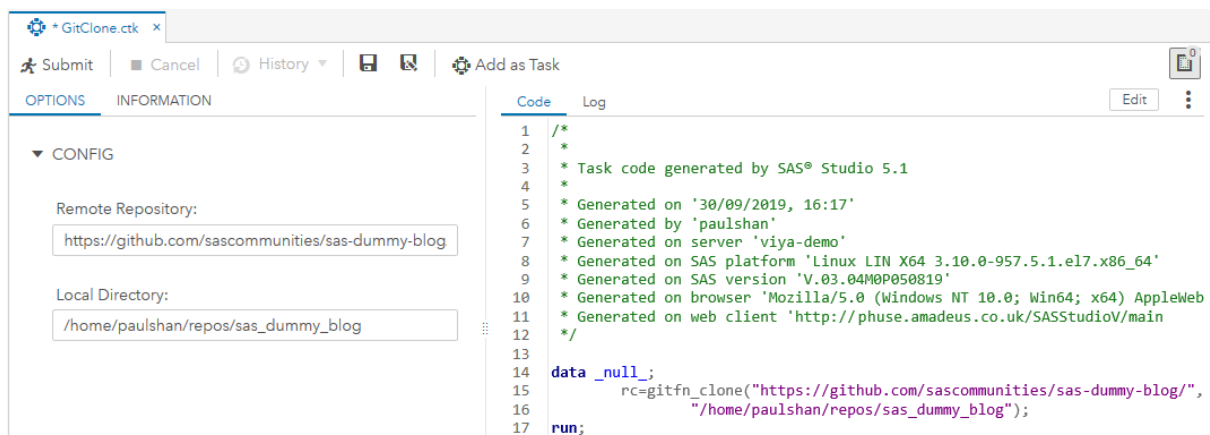


Figure 19 - Using a Custom Task to Clone a Git Repository

In the above example, we have created a custom task with two input fields in the user interface, these are used to populate the required fields in the `gitfn_clone` function. The task generates the necessary `data _null_` step, that will create a clone of the specified Git repository.



CONCLUSION

The design of SAS Studio's user interface is instantly familiar to both users of SAS Enterprise Guide and SAS Display Manager. Favourite features of both these client applications are carried-over and/or enhanced in SAS Studio.

The architecture of web applications means they are often favoured by IT administrators, providing measurable efficiency and cost benefits to any platform administrators. Such architecture is faster and simpler to deploy, maintain, upgrade and ensures greater consistency across a company.

As more organisations migrate to cloud hosted infrastructure and platforms, SAS Studio dependably offers a consistent programming environment that can easily scale to fit the performance needs of companies large and small.

SAS Enterprise Guide continues to be developed; the recent release of v8.1 demonstrates an upgraded user interface that starts to align towards that of SAS Studio, (and other client applications). It's feasible that over time, Enterprise Guide will gradually become a SAS Studio store app, offering cross-platform compatibility on devices of all types, including mobile and touchscreen devices. This may seem to contradict the rest of this paper; however, store apps still offer a single point of deployment (albeit separately for Android, iOS and Windows devices), with updates managed by vendor rather than internally and can still offer user interface benefits too.

REFERENCES

SAS Help Center: Whats New in SAS Studio 3.8, visited on 30 Sep 2019.

<https://documentation.sas.com/?activeCdc=webeditorcdc&cdcId=sasstudiocdc&cdcVersion=3.8&docsetId=webeditorug&docsetTarget=whatsnewwebeditor37.htm&locale=en&docsetVersion=3.8>

SAS Studio 5.1: User's Guide, visited on 30 Sep 2019.

<https://documentation.sas.com/api/docsets/webeditorug/5.1/content/webeditorug.pdf?locale=en>

My Enterprise Guide, visited on 30 Sep 2019.

<https://amadeus.co.uk/assets/Uploads/My-Enterprise-Guide-v1-2.pdf>.

SAS Studio 3.5 Developers Guide to Writing Custom Tasks, visited on 30 Sep 2019.

<https://support.sas.com/documentation/cdl/en/webeditordg/68829/PDF/default/webeditordg.pdf>

Using built-in Git operations in SAS, visited on 30 Sep 2019.

<https://blogs.sas.com/content/sasdummy/2019/01/17/git-in-sas/>

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Paul Shannon

Amadeus Software

The Old School Hall, Wesley Walk, Witney, Oxfordshire, UK. OX28 6ZJ

+44 1993 848 010

paul.shannon@amadeus.co.uk

<https://amadeus.co.uk>

Brand and product names are trademarks of their respective companies.