

How to Learn Software

Our world is becoming ever more technology driven, and it will be increasingly important to have the ability to learn new software skills to stay competitive and keep up with the pace of modern life. Software is the application layer of a computer system, and is the part of the computer system that individuals interact with directly to do whatever task they have at the time.¹ At this point there is a wide variety of tasks and types of work that can be aided or completed entirely with a computer system and the appropriate software, so most people are likely to need to use computer software for at least a portion of their work and leisure activity, and therefore knowing effective strategies to learn software is important.

This paper will examine software specifically in the area of music, where software can be used for many different functions. In Martin Russ' book "Sound Synthesis and Sampling", he notes that:

Computers have almost integrated music totally. Modern music-making software tends to use words such as "Digital Audio Workstation (DAW)" or "Music Sequencer". It is remarkably easy to buy and install a piece of software that has a sequencer, sampler, analogue and digital synthesizers, effects units, mixer, samples of real instruments and drum sounds, and tutorials on how to use it, and which costs a fraction of what exactly the same equipment in real physical form would have cost 10 years ago. This is an astonishing achievement.²

This paper will not look at DAWs per se, but this does show how computers are handling ever more tasks in music. Instead, this paper will look at the use of software for music notation, particularly the notation program³ Dorico produced by Steinberg, as well as synthesizer software

¹"Software and its Types", Geeksforgeeks, Last modified August 29, 2023, <https://www.geeksforgeeks.org/software-and-its-types/>

² Martin Russ, *Sound Synthesis and Sampling*, 3rd edition (Oxford: Focal Press, 2009), 720.

³ Note that the terms "software" and "program" may be used interchangeably in this paper.

that can be used to produce electronic sound (and which can be used in a DAW as mentioned in the quote), in this case Native Instruments FM8. After the general outline of software learning methods at the beginning, this paper will include two case-studies of methods and sources of learning and teaching software in practice.

One technique to learn a program is to become familiar with the overall operation of the program, often through reading a manual or watching certain kinds of videos. This would involve learning the layout of the user interface, what options are available in different menus and dialog boxes in the software, and “what all those buttons do”. For the purposes of this paper, this will be called a “function-based approach”⁴. This approach can be effective, but it has its limits as it doesn’t put anything in context, so it can be difficult to know what will actually be useful when using the tool in a project context. This is particularly an issue with a complicated program like notation software, which often has many different potential applications; for instance creating scores and parts for film score recording sessions⁵ or for concert music⁶, which have somewhat different needs. If one tried to learn every potential function of the program, then not only would one waste time learning functions that may not be applicable for their use case, but it may still not be clear what exactly one would need to do to create any particular type of project.

This approach can be more effective when used with something like a synthesizer such as Native Instruments FM8, where the software is somewhat simpler than notation software, and there are fewer discrete potential applications. Everyone using such a program will be using it to generate sound of one form or another, either for music production, sound design and effects, or

⁴ These method names have been coined by the author for this paper.

⁵ Dorico, “Alan Silvestri, Scoring a Blockbuster, Part 1: From Cubase to Dorico | Artist”, Youtube video, 7:11, April 26, 2019, <https://www.youtube.com/watch?v=GrDqGEQux7g>

⁶ Dorico, “Arranging in Dorico Pro with Septura Brass Septet | Artist”, Youtube Video, 5:23, May 30, 2018, <https://www.youtube.com/watch?v=KqIKN4-CuVw>

possibly sonic education (the author of this paper has used FM8 to explore certain sonic illusions, for example, such as the “missing fundamental” phenomenon). But all of these potential applications will use similar functions and therefore an approach that looks at every function and where it is located and some potential applications will be a useful place to start with such a program.

Another common approach to learning software is a “project-based approach”⁷. In this case, one would have a particular project in mind, which could be a real project for a class or a client, or might be made up just for the sake of trying the software, and one would then try to figure out how to go about creating the end product. Many training resources will include components aimed at setting up projects for the student to complete, as will be discussed in both of the case studies.

The project-based approach is also aided a great deal today by search engines such as Google, and online documentation (user manuals) and forums where people can ask a question (which can then be indexed by the search engines). This will be discussed more below in the case study on Dorico. Steinberg in particular has a great online manual system for Dorico and their other software, which has been fully indexed by Google, and often has task-based, if not project-based, guides for common operations with step-by-step instructions⁸.

These two techniques, the function-based approach and the project-based approach, can often be used most effectively in tandem. It will be easier to figure out how to operate a program for a particular project when one has a basic understanding of the layout of the software, and a broad knowledge of possible parameters available in the software, and then once one begins

⁷ Author’s coinage.

⁸ “Dorico Help”, Steinberg, Accessed December 15, 2023.
https://steinberg.help/dorico_pro/v5/en/

using the software in real-world applications, it will become clear what tasks need to be accomplished and one can then dive deeper into the software for their own use case.

The third method that will be discussed in the case studies will be called the “theory-based approach”⁹. This is not so much a technique for learning a specific piece of software, as it is the necessary background to understand why one would use a particular piece of software, or a specific function in a piece of software. Sometimes this theory will be discussed directly in the material for the software, and other times it is assumed that one knows the theory in advance of using the software, particularly when it comes to user manuals. Video courses (or other formal training programs) will often include some of the theory, as will be shown below in relation to FM8 by Native Instruments.

Case Study No. 1 - MacProVideo’s Course for Native Instruments FM8

One website that hosts quite a few videos for software training is called MacProVideo.com. This website focuses on training for audio, video production, and graphics software. This section will examine in detail one video course from the website, David Earl’s video series for Native Instruments’ frequency modulation (FM) synthesizer software, FM8¹⁰.

This video series consists of several relatively short videos covering a variety of topics related to FM8. The shortest video is 2 minutes and 34 seconds, and the longest one is 6 minutes and 35 seconds. There are 17 videos total and a total running time for the series of 1 hour and 27 minutes. The short videos make it possible to find a topic you are interested in if you have a specific question, as well as making it easy to watch the series in short, bite-sized chunks rather

⁹ Author’s coinage.

¹⁰ David Earl, “FM8: FM Synthesis and Sound Design”, MacProVideo, Accessed December 13, 2023, Online Video Course, <https://macprovideo.com/course/native-instruments-210-fm-synthesis-sound-design>.

than having to spend a long time in a given session if the student does not have a lot of time at any given point in time.

This video series has examples of all three different ways of learning software programs discussed above. The first big part of the series (videos 2 - 9) are a function-based look at the interface of the FM8 software, going through each section of the basic interface, and then looking at the different pages in the main section of the interface. In these videos, Earl covers what the buttons and sliders and other interface elements do in each section of the software in full detail.

Then in the second big part of the video series (videos 10-12, and then also videos 16 and 17 in particular), Earl switches to a more project-based model. In these videos he examines different types of sounds that are common in FM synthesis, including brass sounds¹¹ and bell sounds¹², as well as the way that FM8 in particular allows one to generate simple, 6 oscillator additive synthesis sounds.¹³ At the end of the course, he then dives deeper into more esoteric sound design techniques using FM8, including both how to set up rhythmic patterns in the synthesizer¹⁴, and general sound design practice at the end¹⁵.

In between the project-based videos, he has a couple of videos demonstrating how to use the envelope generators in FM8. These videos are the primary theory-based videos, exploring how envelopes work in synthesis generally in addition to how to specifically use the envelopes in FM8. He also takes some of this approach in the first video in the series where he introduces the synthesizer and discusses some of the history of FM Synthesis¹⁶. This is not really required

¹¹ Earl, "FM8...", Video No. 11

¹² Earl, "FM8...", Video No. 12

¹³ Earl, "FM8...", Video No. 10.

¹⁴ Earl, "FM8...", Video No. 16.

¹⁵ Earl, "FM8..." Video No. 17.

¹⁶ Earl, "FM8...", Video No. 1.

information to learn how to do something with the software, but it will enhance the experience of using it and might give the user some ideas of what they could do with the software, in addition to just adding some interesting context to everything else.

This video series by David Earl for MacProVideo is a good demonstration of several models of teaching software skills, showing function-based, project-based, and theory-based approaches to the topic. This makes the video series a useful place for a person to start learning how to use this software to make interesting sounds, going beyond the function-based approach that is found in the software manual¹⁷ (which mostly goes through the interface elements but with no task-based (project-based) methods, and not much in the way of theory-based methods), to showing real-world applications that make the software feel more immediately useful, especially if one has not used an FM synthesizer before.

For more theory-based elements, there are many useful books about synthesis techniques in general and FM synthesis in particular. The Martin Russ book referenced above has a good section on FM synthesis¹⁸, and another book about FM Synthesis in particular is “How to Make a Noise: Frequency Modulation Synthesis” by Simon Cann, which also has some ideas for project-based learning and specifically explores Native Instruments FM8¹⁹.

Case Study No. 2: Learning Steinberg Dorico Notation Software

The author of this paper has switched from using Avid Sibelius notation software (which he used for about 15 years prior to this) to using Steinberg Dorico notation software over the past

¹⁷ Thomas Loop, *FM8: Operation Manual* (Los Angeles, Native Instruments, 2006), PDF.

¹⁸ Russ, *Sound Synthesis and Sampling*, Section 5.1, 496.

¹⁹ Simon Cann, *How to Make a Noise: Frequency Modulation Synthesis* (Surrey, UK: Coombe Hill Publishing, 2011), Kindle.

several years, and he has mostly taught himself how to use Dorico with the help of several free internet-based resources prepared by or run by Steinberg to help teach people the new software.

Often when one is trying to learn another software program of a similar type to one that they already know how to use, learning the new software is not too difficult, as the main thing that changes is what the menus and functions are called and where they are placed on the interface. Switching from Sibelius to Dorico was somewhat more complicated, because the software's model of how notation should be thought of is quite different to Sibelius' model, so when using Dorico one has to get used to a rather different way of working. The Dorico team had a development blog while they were working on the first version of the software where they explored the model they were developing and how it differed from other software in some detail.²⁰ One simple example of this is the way that the two programs handle a quarter note tied to a quarter note (across a barline for example). Sibelius will treat that as three objects, and if one were to move them to another location, they would remain in that configuration. Dorico thinks of this as a note that is four eighth notes long, and will change the appearance to a half note if you move that unit to the beginning of a measure (for example)²¹. Dorico is full of little things like this that make it somewhat more difficult to learn if one is coming from Sibelius (or most other notation software), but that are often very helpful if one can learn to think more like Dorico does.

Steinberg has put out quite a bit of their own material making use of a variety of training techniques. They have function-based approaches to teaching the layout of the interface in both

²⁰ Daniel Spreadbury, "The First Five Months", *Making Notes* (blog), Steinberg, April 5, 2013, <https://blog.dorico.com/2013/04/the-first-five-months/> - This is the first of many blog posts on the Dorico blog that explores their development process for Dorico. The rest have "Development Diary" in the title of the posts.

²¹ "Rhythmic Position", Dorico Help, Steinberg, Accessed December 15, 2023, https://steinberg.help/dorico_pro/v5/en/dorico/topics/program_concepts/program_concepts_rhythmic_position_c.html

the manual²² and videos that they have released on a quite active YouTube channel²³. They have an ongoing series on the channel called “Discover Dorico”, which consists of a series of livestreams where they examine everything from new features of updates²⁴ (often function-based) to how to input particular types of music most effectively²⁵ (project-based).

Another project-based tool is the Google-indexed manual mentioned above, which allows one to work on a project of their own and find quick answers to questions about how to do something that they need to do at that time. If one types in the question to Google, then frequently the manual is one of the first results, which gets pulled right to the page that explains in a step-by-step manner how to do that task. If the manual doesn’t have a result for that, then often someone else will have asked that question at some point on the Dorico forums, which are another resource that usually pops up in the first few search results.²⁶

The theory-based approach to learning Dorico would be to study notation practices in general, and maybe the history of music engraving or how traditional publishing houses have set various scores and parts over the centuries. Steinberg’s material doesn’t dive too much into that (though the development blog does cover quite a bit of this material throughout the entries²⁷), but there are plenty of other books available on the topic. One of the major reference sources today is

²² “Project Window”, Dorico Help, Steinberg, Accessed December 15, 2023, https://steinberg.help/dorico_pro/v5/en/dorico/topics/user_interface/user_interface_windows_c.html

²³ Dorico, Youtube, accessed December 15, 2023, <https://www.youtube.com/@dorico>

²⁴ Dorico, “Discover Dorico 3.5 Music Notation Software: Smarter, Faster, Better”, Youtube Video, 1:25:30, January 21, 2021, <https://www.youtube.com/watch?v=q6AUsNm1q80>

²⁵ Dorico, “Create a Title Page”, Youtube Video, 58:30, November 29, 2023, https://www.youtube.com/watch?v=il0ATZ_9UyE

²⁶ Dorico, Steinberg Forums, accessed December 15, 2023, <https://forums.steinberg.net/c/dorico/8>

²⁷ Daniel Spreadbury, “Development Diary, Part 6”, *Making Notes* (blog), Steinberg, March 12, 2014, <https://blog.dorico.com/2014/03/development-diary-part-six/> - for just one of several examples.

Elaine Gould's 2011 book "Behind Bars: The Definitive Guide to Music Notation"²⁸. Two other books the author of this paper likes are from Berklee Press (at Berklee College of Music, where the author earned his bachelor's degree): Matthew Nicholl and Richard Grudzinski's book "Music Notation: Preparing Scores and Parts"²⁹, and the more recent book "Berklee Contemporary Music Notation"³⁰ by Jonathan Feist, which covers general notation principles and also dives into genres outside of classical music in a little more detail than some of the other books do.

As shown in this paper, there are several methods that one can use to learn music software: "function-based", "project-based", and "theory-based" approaches. Each of these has their pros and cons and using them in tandem will generally produce better results than using any of them individually will. They can be used either as part of a formal video course or class, or on one's own to teach themselves the software.

This paper can also be used as a guide for developing courses or classes to teach software, as the techniques one would use to learn software are pretty much the same techniques that one would use to teach it to someone else either in a one-on-one context (e.g. tutoring), or in a formal classroom setting. This paper examined a video course for Native Instruments FM8 in the context of using it to teach oneself the software, but when David Earl made the series, he would have had to take the same things into account to structure it in the first place. These skills will continue to come in handy as new software is released and old software is no longer maintained, requiring the constant learning of new software over time.

²⁸ Elaine Gould, *Behind Bars: The Definitive Guide to Music Notation* (London: Faber Music Ltd, 2011).

²⁹ Matthew Nicholl and Richard Grudzinski, *Music Notation: Preparing Scores and Parts*, Boston: Berklee Press, 2007.

³⁰ Jonathan Feist, *Berklee Contemporary Music Notation*, Boston: Berklee Press, 2017.

Bibliography

- Cann, Simon. *How to Make a Noise: Frequency Modulation Synthesis*. Surrey, UK: Coombe Hill Publishing, 2011. Kindle.
- “Dorico Help”. Steinberg. Accessed December 15, 2023. https://steinberg.help/dorico_pro/v5/en/
- Dorico. Steinberg Forums. Accessed December 15, 2023, <https://forums.steinberg.net/c/dorico/8>
- Dorico. Youtube. Accessed December 15, 2023. <https://www.youtube.com/@dorico>
- Earl, David. “FM8: FM Synthesis and Sound Design”. MacProVideo. Accessed December 13, 2023. Online Video Course. <https://macprovideo.com/course/native-instruments-210-fm-synthesis-sound-design>.
- Feist, Jonathan. *Berklee Contemporary Music Notation*. Boston: Berklee Press, 2007.
- Gould, Elaine. *Behind Bars: The Definitive Guide to Music Notation*. London: Faber Music Ltd., 2011.
- Loop, Thomas. *FM8: Operation Manual*. Los Angeles, Native Instruments, 2006. PDF.
- Nicholl, Matthew and Grudzinski, Richard. *Music Notation: Preparing Scores and Parts*. Boston: Berklee Press, 2007.
- Russ, Martin. *Sound Synthesis and Sampling, 3rd Edition*. Oxford: Focal Press, 2009.
- “Software and its Types”. Geeksforgeeks. Last modified August 29, 2023. <https://www.geeksforgeeks.org/software-and-its-types/>
- Spreadbury, Daniel. *Making Notes* (blog). Steinberg. <https://blog.dorico.com/making-notes/>