

# Section 1

## Getting Oriented

Cambridge University Press  
978-1-108-73836-1 — Excel Basics to Blackbelt  
3rd Edition  
Excerpt  
[More Information](#)

---

# 1

## Necessary Foundations for Decision Support

**Objective:** *Appreciate the wide range of intentions, focal methods, and general characteristics of effective decision support tools.*

We make decisions all the time. Some decisions may appear routine: What shirt to wear? What to have for lunch? Should I purchase this online or go to the store? Should I respond to a post that I feel knowledgeable about? Others are more complex: Should I recommend that my client invest in a particular firm? Should I offer to take on additional work? Should my firm build or buy a new technology to help with its processes? Should I recommend a settlement in a lawsuit?

With complex questions comes the need for complex consideration. Sometimes we need help with these considerations. The sources of help can vary, but increasingly these sources tend to have two things in common: facilitated analytical strength and facilitated access – access to both underlying data and processes as well as to means by which to convey results to others. These sources of assistance often take the form of prepackaged off-the-shelf software tools. However, they can also be uniquely customized, and more and more frequently, this customization is being developed by individual users.

### 1.1 Intentions and Approaches

We can begin to understand the universe of tools designed to assist with decision making by listing their potential benefits. A few of the more common appear in Figure 1.1. For many managers, developers, and analysts, only a few of these sample benefits will prove essential. For others, the full complement of potential benefits must be considered along with others. However, the omission of one of these benefits should not be viewed as a failure. Similarly, not all efforts to try to squeeze each of these into a single tool are justifiable. Like all things, there are pros and cons to design

*Typical Attributes*

**Eased Access** (to raw distributed data often updated in near-real time)

**Facilitated Analysis** (of data often through use of automated intelligence)

**Rich Communication** (of results and new ideas in meaningful and practical form, often augmented by sophisticated graphical depictions)

*Common Benefits*

**Reduced lead time**  
(to complete work)

**Greater Consistency**

**Smarter Response**  
(to changes / failures)

**Worker empowerment**

**Reduced cost**

**Increased Innovativeness**

**Greater partner satisfaction**  
(both customers and suppliers)

**Higher Retention**

**Elevated Strategic Advantage**

Figure 1.1 Common elements of DSS

complexity. Tools that are not sufficiently straightforward won't get used. Tools that include too many features may not see timely completion; and since an excess of features can detract from users' ability to find key features or to leverage principal functionality, again such tools might not get used. This defeats the core purpose of Decision Support Systems (DSS): to support decision making.

So, let's back up a second and think somewhat differently about this. The design of the fundamental attributes that are typical of all DSS, facilitated analytics and access, must be deliberately aligned to desired benefits in order for those benefits to arise. And, if we need to be somewhat cautious, pragmatically, regarding the variety of benefits we are shooting for in a DSS design, then we can and should be specific about these attributes as well.

That shouldn't be a surprise to anyone. If you have a specific objective to accomplish, you do well to use the most appropriate approaches. The same general rule applies to all the forms of analysis that are facilitated by a DSS. Broadly, for example, we tend to delineate between descriptive, predictive, and prescriptive analytics, each and/or all of which might support a particular objective benefit in decision support. Descriptive work can provide details of the rich landscape in which decisions are made, even if the possible interconnections within that landscape have not yet been verified. Statistically we can describe measures of centrality such as means and medians, measures of variation and its possible skew. Computationally we can describe the multi-dimensional landscape visually, we can filter, or we can aggregate and

explore it interactively. Descriptive approaches built into tools can go a very long way in supporting decision making, though they may not go far enough.

Taking this one level further, our needs may require us to develop tools that are predictive, attempting to justify interconnections we may have begun to suspect through our descriptive statistical and computational efforts. Once again, we might approach this task using established statistical analyses to verify significant relationships between variables in our data, or we could leverage more computationally intensive and fluid methods such as stepwise estimation, variations on network analysis, or machine learning. A level yet further would involve prescriptive analytics to attempt to determine closed-form statistical expected value maximizing decision sets, or computationally searching for optimal or theoretically near-optimal ones.

How we pursue any one of these efforts, from data description through prescription, is dependent on the decision-making context in question and, once again, the objective benefits targeted. If we are able to begin with a fairly clear vision of what we need to achieve, or are at least willing to return to reevaluate this vision, we are going to be far more likely to make the right choices at each step of this journey.

## **1.2 Stages in Development**

With this perspective in mind, we can also start to break down the various stages across which an integrated decision support system, potentially relying on all three levels of analysis, might be developed. Figure 1.2 provides a general map of these stages in a complete process that recognizes that forward progress can easily lead to a return to earlier stages for reconsideration. For a related decision tree depiction developed for the International Institute for Analytics see [www.excel-blackbelt.com](http://www.excel-blackbelt.com).

Across these stages we can envision a host of opportunities in which to use any number of smaller tools, functions, and subroutines to help us get the job done. Some of these tactics are executed once and seldom returned to. Others are functions that respond dynamically to new information and changes in other aspects of our approaches. Some are point and click, others require a bit more typing and organizing, while still others involve some serious research and not a small amount of trial-and-error to yield results. As we will see in the chapters of this book, there is a great deal that we learn just by trying; we gain a much deeper understanding of our options for future development, as well as pitfalls we might sidestep.

Among these stages we also find numerous opportunities for visualization, encompassing a range of analytical tactics, to contribute value. Indeed, the application of visualization in data analysis is critical to the development of decision support systems. It grants individual developers a view into what

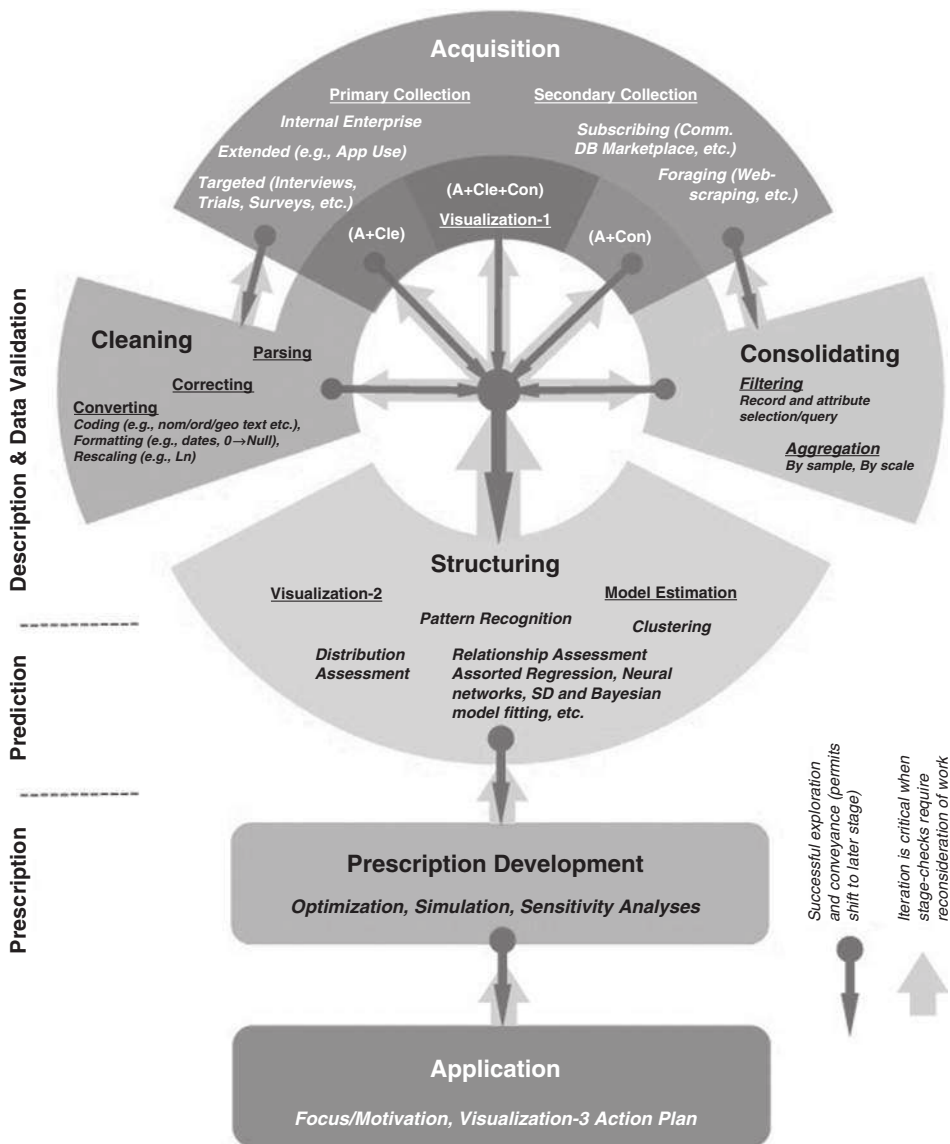


Figure 1.2 Stages in DSS development

opportunities and flaws exist in their work and assumptions, but it also provides the means to translate descriptions, predictions, and prescriptions into meaningful messages for various audiences served by these tools and their results. By doing so, it supports feedback that can further enhance the DSS itself.

Consider the following key principles described in *Beautiful Evidence* by Edward Tufte, a recognized scholar on data and relational visualization:

- 1) *Enforce Wise Visual Comparisons*: Comparison is a critical element in the development of understanding regarding the features of and anomalies within data.

Therefore, it is also critical in the practical application of findings from analysis. Comparison allows for the illustration of practical relevance of effects and decisions that may give rise to them. Wise visual comparisons are the mechanisms by which analytical and theoretical findings are vetted by real-world experience. They encourage faith in the analyst, and hence in most systems, as well as in the framework that the analyst is recommending.

- 2) *Show Cause*: Most experienced and practical researchers cringe when individuals confuse mere statistical correlation with causality. These assertions often lack evidence. However, in some cases certain reasoned explanations can be convincing. The task of the developer is to provide his or her reader with enough temporal and situational information for causal suggestions, either stated or implied, to be clear. The developer should also be able to facilitate scrutiny of data and analysis where appropriate. Tools should ultimately allow audiences to draw their own intelligently structured conclusions regarding causality rather than portray the perspectives of developers as unassailable.
- 3) *The World We Seek to Understand Is Multivariate, as Our Displays Should Be (AND as our analysis that supports them should be)*: This is the classic story of the blind men and the elephant. Considering one dimension of data gives only limited and potentially flawed understanding of the big picture. However, not all pieces of information are useful. A little bit of rationality goes a long way in analysis. Hence one should push for multidimensional analysis and visualization to satisfy the needs of sufficiency, while being wary of distraction.
- 4) *Completely Integrate Words, Numbers, and Images*: Inconsistency between prose, data, and graphics weakens arguments. At the very least it confuses, at worst it misinforms. Analysts need to view graphical, numerical, and textual content as reinforcement mechanisms complementing each other. When alternate perspectives can be presented, don't confound these distinctions with differences in conveyance. Staying true here is another key to generating faith in the analyst and his or her tool.
- 5) *Most of What Happens in Design Depends upon the Quality, Relevance, and Integrity of Content*: Garbage fed into a graph results, at best, in beautiful garbage (but it still stinks). Know your audience. Understand the practice. Be fully aware of what needs to be analyzed and do it the right way. Mistakes and irrelevant detail can weaken your work.

Regardless of the technical nature of decision support tools, a serious consideration of these principles during development helps guarantee future use. While we will not devote a great deal of time to the psychological responses individuals have to certain visual forms, and hence best practices in visual designs, we will examine a range of technical approaches and continue to reiterate the importance of thought put into their selection and use. A more in-depth discussion of best practices in visual designs, backed by psychological theory and recent research, can be found in *Visual Analytics for Management* (2017).

Aside from these recommendations, it is essential to emphasize another point here, particularly for those new to DSS development: Decision support systems refer to applications that are designed to *support*, not *replace*, decision making. Unfortunately, DSS users (and developers) too often forget this concept, or the users simply equate the notion of intelligent support of human decision making with automated decision making. Not only does that miss the point of the application development, but it also sets up a sequence of potentially disruptive outcomes. These include excessive anthropomorphism, poor or impractical decisions, disastrous results, and the scapegoating of IT technicians. *It's easy for decision makers to view decision support systems as remedies for difficult work, particularly if they can blame others when things don't work.*

Although it is often difficult to codify, there is an implied contract between those who claim to deliver intelligent tools and those who accept their use: namely, an agreement that the analyst will not attempt to intentionally mislead the user. Ultimately, these issues contribute to the accountability and ethics of organizations, as well as the personal accountability of those developing the applications. If you want to develop a strong decision support tool, you have to identify your desk as where the buck stops. But if you want to be able to share the tool, you have to pay attention to how your applications might be used by others. It is critical here to guide built-in assistance in analysis, clear visualization of how characteristics of problems and solutions relate, and formally structured interfaces that deter, if not prohibit, misuse. It can be difficult to distinguish between those who intend to use their positions dutifully, but fail in execution, from those who will use their positions to deter others. If you set out to provide support for the right reasons, make sure you provide that support the right way.

### 1.3 Overview and Resources

Unlike many texts on related topics, this book is prepared for both the developer and the ultimate user. When reading these chapters, readers will learn what to expect from DSS and from those already using and building DSS tools. Fundamentally, it should become clear that the creation of decision support tools is something that even individuals with very little programming skill (or interest) can accomplish. That having been said, it will also be made clear that a little more programming acumen can help you accomplish a great deal more. This book works to drive home these points and to emphasize that individuals with just a bit of DSS development experience under their belts become popular very fast. They serve as bridges between career programmers who can build highly sophisticated resources, and the managers who face the reality in which only certain specific designs are going to prove useful.



How do we get there? We will need to acquire some experience in the various stages of development outlined in Figure 1.2, and we will need to do that in a way that is accessible (there's that word again) to essentially anyone. Ideally, we'll use a platform that is also still a backbone resource at most organizations, even if it's one that is often misperceived by those who know little about or overgeneralize its potential. We have that in the highly ubiquitous, deceptively familiar Microsoft Excel environment. Complementing this environment in our journey will be the use of associated tools, add-ins, code development, and affiliated applications such as the Microsoft BI suite. We will also discuss connections to other software environments such as those provided by Google, Palisade, and Apple.

What is the value of constructing tools using resources that career programmers might not champion in the development of commercialized applications? The question practically answers itself: Learning. If you have no intention of becoming a career programmer but would just like to learn how to get decision support tools built, and wouldn't mind becoming a bit more influential along the way, you need a ramp. The environments we will be discussing, and the approaches used within them, provide that ramp. You'll have the ability to get your hands dirty in each stage of development, from soup to nuts. You'll gain a systems perspective of how systems are built and how they work. Much like acquiring logic skills through programming experience, these skills are highly transferable. Further, the proof-of-concepts that you end up building can provide talking points and launch pads for further development – positioning you squarely to consult on the most effective paths that career programmers should take.

To help us through discussions and exercises in development, this book uses simplified contexts from real-world practice to enrich instruction on decision support. For example, we will introduce the case of a restaurant struggling to reconsider how to redesign itself in the face of changes in preferences. The Lobo's Cantina case gives us an opportunity to examine data on which predictions are built and the foundation of optimal policy and resource allocation decisions are made. How much space should be allocated to certain kinds of customer seating in the rebuild? How should we manage overbooking policies, given observed uncertainties in demand and human behavior? What kinds of risk profiles do our options in these decision-making contexts reveal, and how do we ultimately select the options to go with?

In another case to which we will return at multiple points in the book, we will consider the decisions of a firm trying to organize the shipping and transshipment of goods across national boundaries. Once again we start with discussions what we might do with data acquired through the firm's experiences. How can we describe visually the complexity of the tasks at hand? How can predictions in this context inform prescriptions? How do we

make choices in the presence of expected levels of risk, where are the safe bets, what are the next-best options, and how do we document and convey the anticipated returns and risks associated with the best options we settle on?

As for the structure of this book, we will generally follow the forward flow presented in Figure 1.2. This first section, titled “Getting Oriented,” focuses on easing people into the Excel workbook environment as a platform for tool development and visualization. Navigation and data acquisition are central themes, as are discussions of cleaning and consolidation tactics, along with critical caveats regarding some of these approaches. In this and other sections of the book we will consider some of the most valuable and often underused of Excel’s functions and tools. We will also introduce the Blackbelt Ribbon add-in and the additional capabilities it provides.

In the second section, “Structuring Intelligence,” we will begin to discuss what acquired data can tell us about the real-world contexts in which we must make our decisions. We will briefly consider how predictions of trends and relationships in data might be estimated but will pay special attention to the uncertainty that these predictions can imply. With this in mind we will describe the design and construction of simple simulations based on such real-world analysis. We will consider how to jointly leverage tools such as data tables in tandem with consolidation tactics such as Pivot Tables in the design of these assessments as well as the use of user-friendly controls to simplify the management of this work. The visualization of available data and the by-products of simulation will then be discussed. The focus will be on specific tools for the workbook and related environments, with reference to best practices in design outlined in texts such as *Visual Analytics for Management* (2017). Here we will also get some additional glimpses into the world of programming.

In the third section, “Prescription Development,” we forge ahead into discussions of the construction of mathematical and functional architectures for arriving at specific recommendations in decision making. We focus on available tools but also push beyond common experience with these tools. We will discuss how we might peek behind the optimization approaches used by tools such as Solver, as well as approaches made available by Palisade in their @RISK toolset. We will discuss limitations of various approaches, the integration of optimization and simulation tactics to seek out both high expected value and low-risk options, and critically consider the trade-offs between search time, search progress, and the practical value of benefits gained.

In the fourth section, “Advanced Automation and Interfacing,” we dive more deeply into development work that could not easily be conducted in the spreadsheet environment alone. Instead, we will turn our focus to the Visual