O'REILLY®

Business Models for the Data Economy



Q Ethan McCallum
& Ken Gleason

www.dbooks.org



SAN JOSE



LONDON



NEW YORK



SINGAPORE

Strata+ Hadoop WORLD

Make Data Work

strataconf.com

Presented by O'Reilly and Cloudera, Strata + Hadoop World is where cutting-edge data science and new business fundamentals intersect—and merge.

- Learn business applications of data technologies
- Develop new skills through trainings and in-depth tutorials
- Connect with an international community of thousands who work with data

Job # 15420

Business Models for the Data Economy

Q Ethan McCallum and Ken Gleason

O'REILLY®
Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

www.dbooks.org

Business Models for the Data Economy

by Q Ethan McCallum and Ken Gleason

Copyright © 2013 Q Ethan McCallum and Ken Gleason. All rights reserved.

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://my.safaribooksonline.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Editor: Mike Loukides

October 2013: First Edition

Revision History for the First Edition:

2013-10-01: First release

Nutshell Handbook, the Nutshell Handbook logo, and the O'Reilly logo are registered trademarks of O'Reilly Media, Inc. *Business Models for the Data Economy* and related trade dress are trademarks of O'Reilly Media, Inc.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and O'Reilly Media, Inc., was aware of a trademark claim, the designations have been printed in caps or initial caps.

While every precaution has been taken in the preparation of this book, the publisher and authors assume no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.

ISBN: 978-1-449-37223-1

[LSI]

Table of Contents

Business Models for the Data Economy **1**
Collect/Supply 2
Store/Host 3
Filter/Refine 5
Enhance/Enrich 8
Simplify Access 9
Analyze 10
Obscure 12
Consult/Advise 14
Considerations 15
  Domain Knowledge 15
  Technical Skills 16
  Usage Rights 18
  Business Concerns: Pricing Strategies, Economics, and Watching the Bottom Line 18
Conclusion 20

www.dbooks.org

Business Models for the Data Economy

Whether you call it Big Data, data science, or simply analytics, modern businesses see data as a gold mine. Sometimes they already have this data in hand and understand that it is central to their activities. Other times, they uncover new data that fills a perceived gap, or seemingly "useless" data generated by other processes. Whatever the case, there is certainly value in using data to advance your business.

Few businesses would pass up an opportunity to predict future events, better understand their clients, or otherwise improve their standing. Still, many of these same companies fail to realize they even *have* rich sources of data, much less how to capitalize on them. Unaware of the opportunities, they unwittingly leave money on the table.

Other businesses may fall into an explore/exploit imbalance in their attempts to monetize their data: they invest lots of energy looking for a profitable idea and become very risk averse once they stumble onto the first one one that works. They use only that one idea (exploit) and fail to look for others that may be equally if not more profitable (explore).[1]

We hope this paper will inspire ideas if you're in the first camp or encourage more exploration if you're part of the second so you can build a broad and balanced portfolio of techniques. While there are

1. See the first chapter of John Myles White's *Bandit Algorithms for Website Optimization* for a brief yet informative explanation of the explore-versus-exploit conundrum. Clayton Christensen also explores this concept in *The Innovator's Dilemma* (HarperBusiness), though he refers to it in terms of innovation instead of algorithms.

www.dbooks.org

myriad ways to make data profitable, they are all rooted in the core strategies we present in the following list.

Collect/Supply
: Gather and sell raw data

Store/Host
: Hold onto someone else's data for them

Filter/Refine
: Strip out problematic records or data fields or release interesting data subsets

Enhance/Enrich
: Blend in other datasets to create a new and interesting picture

Simplify Access
: Help people cherry-pick the data they want in the format they prefer

Obscure
: Inhibit people from seeing or collecting certain information

Consult/Advise
: Provide guidance on others' data efforts

As a frame of reference, we'll provide real-world examples of these strategies when appropriate. Astute readers will note that these strategies are closely related and occasionally overlap.

There are plenty of business opportunities in selling refined data and specialized services therein. In certain cases, the data needn't even be *yours* in order for you to profit from it. While we'll spend most of our time on the more innovative topics, we'll start with the simplest of all strategies: the one we call Collect/Supply, which can stand alone or serve as a foundation for others.

Collect/Supply

Let's start with the humble, tried-and-true option: build a dataset (collect) and then sell it (supply). If it's difficult or time-consuming for others to collect certain data, then they'll certainly pay someone else to do it.

This is hardly a new business. Companies have been collecting and reselling data since before the computerized database was invented.

Just ask anyone who manages subscription lists for magazines. It's hardly sexy, but grunt work sure pays the bills. That's because people will happily trade money for work they don't want to do—or can't do well.

That explains why people will *buy* someone else's data. What's the appeal for someone who wishes to *sell* data? In a word: *simplicity*. You gather data, either by hand or through scraping, and you sell it to interested parties. No fuss, no muss. Unlike with physical goods, you can resell that same dataset over and over. While your cost of creation might be high (sometimes this involves manual data entry, or other work you cannot easily automate), you have near-zero marginal cost of distribution (if you distribute the data electronically). Your greatest recurring expenses should be fees for storage and bandwidth, both of which continue to decline. There's plenty of hard work between the inspiration and the payoff, but efficiency and utter simplicity should be as much a goal as the data itself.

Sometimes you don't have to collect first, as you already have the data. Perhaps it is a byproduct of what you already do. Say, for example, you've developed a new stock market–forecasting model. Along the way, you've collected time-series data from several financial news channels, then made the painstaking adjustments for time misalignment between those sources. Even if your model fails, you still have a dataset that someone else may deem of value.

Collect/Supply is a simple option, and it's certainly one you should consider. As we continue our survey of ways to profit from data, we'll show that it is an important first step for other opportunities.

Store/Host

Store/Host is a subtle twist on Collect/Supply.

People certainly need a place to store all the data they have collected. While a traditional, in-house system or self-managed cloud service makes sense for many businesses, other times it's better to offload management to third parties. This is especially useful for data that is very large or otherwise difficult for clients to store on their own. In essence, they transfer the burden of storage to you. This can be especially helpful (read: *profitable*) when clients are required to store data for regulatory purposes: if you can stomach the contractual burden to

www.dbooks.org

guarantee you'll have the data, clients can rely on you—and, therefore, pay you—to do so and spare themselves the trouble.

As an example, developers can design their apps to send log messages to Loggly. Loggly holds on to the messages as they arrive from various sources—say, handheld applications—and developers can later view the logs in aggregate. This can facilitate troubleshooting a widespread error, or even something as pedestrian as tracking what app versions still run in the wild. Loggly also lets its customers define custom alerts based on conditions such as message content or count. All the while, the developers delegate storage issues to Loggly.

What if you hold the same data for several clients? Here, the economies of scale work in your favor: the marginal cost to store should fall below the marginal value of each client paying you to store it. Case in point: social-media archive service Gnip takes care of collecting and storing data, so their customers can request historical data from Twitter, Facebook, and other sources from them later on. This is very similar to market data resellers, which gather and resell historical tick data to trading shops large and small.

Hosting doesn't have to be just about storing and providing access to raw data. You can also host analysis services: provide your customers with basic summary statistics—or any other calculated measures of the datasets—such that they needn't download the data and do it for themselves. The effort to provide this functionality can range from trivial (simple canned queries) to intricate (freeform queries as chosen by the end user). Consider the TempoDB platform: use it to store your time-series data and also to summarize that data in aggregate.[2] As a second example, customers can stream data straight to BigML and perform freeform modeling and analysis. BigML holds on to the data and runs the calculations on its servers. Google Analytics, the granddaddy of hosted analytics services, is a special case: Google collects and stores the raw, click-by-click data of web traffic on behalf of customers, who then see neat charts and breakdowns.[3]

2. Interestingly enough, it's surprising that the hosted-log services didn't branch off into hosted time-series analysis. Log hosting, seen from a particular angle, is a subset of time-series data hosting.

3. Sharp-eyed readers will note that the Google Analytics example cross-cuts other categories, including Filter/Refine.

Customers derive several benefits from hosted analytics. First of all, they get concentrated expertise in terms of implementation. They also see zero deployment costs as well as development of features they may not have considered ahead of time—as other clients ask you to implement a certain feature, or as you think of new ones yourself, you can roll them out to everyone. Take note: don't confuse a customer's lack of expertise in *constructing* a query with lack of expertise in *validating* the results. Bad analytics can do more damage than none at all, and customers will certainly notice.

The more onerous the calculations, the greater your value-add to your customers, and therein lies the catch: if the calculations are *too* onerous, you risk taxing your systems as customers repeatedly request summary information. This is troublesome enough with canned queries, wherein you can gauge your burdens up front. If you permit ad-hoc analytics across the data (say, through some kind of API or dashboard), then you risk an unbounded set of pain and anguish.

A treatise on how to properly build a performant analytics system is well beyond the scope of this paper. That said, you would do well to have a long think before you turn your storage platform into a general query system. Carefully gauge the required resources in terms of both hardware and bandwidth. Over time, you can determine which queries are common (it's entirely possible that several customers will enter the same freeform query against the same data) and precompute the results. Precomputation will make your customers happy because the results return more quickly and keep you happy because you don't overburden your systems.

Filter/Refine

"Bad data" comes in many forms.[4] One common case is data with malformed, missing, duplicate, or incorrect records. Another business idea, then, is to supply a "clean" dataset that removes or corrects these rogue records. Some people will happily purchase raw data and clean it themselves, yet there are still many more who would rather work on a refined set.

Similar to the Collect/Supply strategy, the Filter/Refine value-add is that you handle the technical grunt work of cleaning the data so your

4. See *Bad Data Handbook* for some examples.

www.dbooks.org

clients can focus on their work. For example, someone who runs a mailing list subscription service will happily pay for someone else to remove duplicates and filter out nonexistent or fraudulent addresses. They don't want to be in the business of managing data; they want to use data to drive their business.

Normalization is another sought-after refinement: the person who buys that mailing list would be very happy for you to disambiguate names and addresses. They would not want to send one mailing to each of J D Doe, John D Doe, and Johnd Doe, all of 123 Main St, when these three names refer to the same person. Similarly, a person researching corporate filings would want to know that GE and General Electric are the same company.

You can also offer downsampling as a kind of filtering. If a client's trading strategies only need hour-by-hour high/low prices, they may prefer to buy that instead of tick-by-tick data that they'll have to sort out on their own.

Selling filtered data can be tricky, though. You'll certainly spend time writing, debugging, and running tools to scrub a dataset, but you'll first have to pass two hurdles:

Defining what is a "bad" record
: Is it simply a record with a missing field? Does a field contain an "incorrect" or even "impossible" value?

Figuring out what to do with a bad record
: Flag it? Remove it? Try to correct it?

These are not easy questions, but their answers lie in two key concepts: *understand the problem domain* and *know your clientele*. Familiarity with the domain will help you understand what constitutes a bad record. An empty field is legitimate in certain contexts, as is a person named John Doe. The restaurant review that complains of space aliens, though, is (hopefully) a joke, and that Chicago home address of 1060 W Addison is unquestionably bogus.

In turn, understanding how your clientele plans to use the data will help you decide how to handle bad records. Someone who is buying an archive of website comments would appreciate that you've already removed the spam content, for example. All in all, the people best suited to sell "clean" data have probably worked in the industry before. Their depth of knowledge in the arena helps them stay ahead of the upstart competition.

Keep in mind, Filter/Refine needn't apply just to static data dumps. Consider real-time or near-real-time data sources, for which you could serve as a middleman between the data's creator and intended recipients in order to filter undesirable records. In other words, your service permits the recipient to build a pristine data store of their own. This would be especially useful for online services that accept end-user input or rely on other external content. Spam comments on a blog send a message that the host is unable or unwilling to perform upkeep, which will deter legitimate visitors. Site maintainers therefore employ spam filters to keep their sites clean. Deeper along the network stack, some routers try to block denial-of-service (DoS) attacks such that the receiving web servers don't crash under the weight of the fraudulent requests.

A novel twist on this concept would confirm the authenticity and timeliness of a news article. A fake story that purports to hail from a reputable service could influence financial markets. Remember the snafu that befell United Airlines stock in 2008? A six-year-old article about the company's financial woes resurfaced, market participants (unaware of the article's age) shorted the stock in an attempt to mitigate their losses, and the rest of the market quickly followed suit.[5] The appropriate news filter may have saved United stockholders—as well as several market participants—from a rather frustrating day.

More recently, and perhaps more disturbingly, someone hacked the Associated Press Twitter account and broadcast a fake headline. The tweet reported explosions at the White House and triggered a sudden drop in the stock market.[6] Granted, this story may have been more difficult to fact-check algorithmically—it was possible that AP was simply first to have reported a genuine event—but it serves as another indicator that businesses rely on social media feeds. They could surely use a service to separate pranks from truths.

5. This story made quite a bit of news. One description is available in *Wired*. At the time of the incident, one of the authors of this paper noticed that the story impacted not just United Airlines stock, but that of several other airlines. Bad news moves quickly and has widespread impact.

6. "A hacked tweet briefly unnerves the stock market"

www.dbooks.org

Enhance/Enrich

Like Filter/Refine, the goal of Enhance/Enrich is to spare people the trouble of preprocessing data on their own. Unlike filtering, however, the strategy here is to *add,* not subtract or normalize. You can create a unique value proposition by joining two datasets, or even deriving some (computationally intensive) results out of a single dataset. You can then sell the enhanced data to other parties, or keep it for yourself to improve your own business.

Combining datasets can prove useful when the join is logically intuitive yet but difficult to perform due to the data's structure or location. Google Maps, for example, continues to integrate new datasets into its geographic data backdrop: restaurants, other businesses, metropolitan transit systems, and so on. It seems obvious once it's there, but this is a classic example of creating value by merging datasets.

Public domain and other open datasets make ideal candidates for Enhance/Enrich operations, as they generally have few restrictions on use. The NTSB Airline On-Time dataset is one such case. It includes a record for every domestic US flight, including each flight's *local* departure and arrival time. For some research, it would be nice to have the *absolute* times (GMT).[7] In this case, simply providing a standardized GMT field could be of value. Even though the calculation is trivial, it's one less step the recipient has to take before performing their own research.

Gauging stock analyst performance provides another example of an Enhance/Enrich exercise. Between the various people, TV stations, newspapers, and even blogs all bellowing stock advice, how do you know whom to trust? One way would be to develop a quantitative measure of the analysts' performance: note the stock price on the day of an analyst's rating change, then compare it to the price some time in the future. Even this basic level of data enrichment could be of value to someone—notably, people who would like to know which analysts are more often wrong than right.

Sometimes you can spare your customers a costly calculation. The volume-weighted average price (VWAP) is a valuable benchmark to institutional portfolio managers who need to buy or sell large blocks

7. We borrowed this idea from OpenFlights.org and an example found in *Big Data for Chimps* (Kromer, Lawson).

of stock spread out over long periods of time. This is a computationally intensive calculation that requires access to market data, since portfolio managers must calculate it uniquely for *every* order on *every* stock they trade. They would certainly value a service that would calculate it for them, rather than having to do it on their own.

Simplify Access

Sometimes companies do not care to deal with raw, bulk data downloads, especially when the data arrives in spreadsheets, fixed-width files, PDFs, or other forms not readily amenable to analysis. That means there's an opportunity for you to give people just the data they need in a format they can handle.

This is a logical extension to an existing Collect/Supply business, with a bit of Filter/Refine thrown on the back end. Given several customers who buy the same raw data from you and who perform the same post-processing, you can do it for them at some reasonable cost.

You don't have to stop there. You can also put the data behind an API such that customers can programmatically fetch the subsets they're after, and in a machine-readable format. Consider cases in which the raw data is both bulky and comprises a large superset of what the customer really wants. They can either download all the data and write their own routines to extract the portions of interest, or they can pay you to subset and extract on demand. Here, the ability to extract specific subsets of the data can be just as valuable (if not more so) than having the entire dataset.

It's often said that most of the effort in a data-analysis exercise is spent on the grunt work: collection, segmenting, subsetting, and cleaning. Researchers appreciate pre-cleaned data because it means they get to dive straight into the analysis and can skip the distraction of prep work.

As an example, consider the NOAA GSOD (Global Summary of the Day) weather dataset. The NOAA provides the data archives for each weather station, by the day or by the year. The raw data files are in fixed-width format. A researcher interested in several years' worth of data for a particular city would have to: determine which weather station(s) report for the city of interest; develop tools to download the data archives for the weather station(s) and time periods of interest; then, develop tools to extract the fields of interest (say, precipitation and temperature) and convert to a more suitable format (say, CSV).

www.dbooks.org

By comparison, what if they could issue a simple REST-style request to a service, specifying the city and date ranges, and receive CSV or JSON data in return? Such is the value-add of Factual.com and similar services.

A simple machine-to-machine, subset-and-reformat service certainly works for tabular data. What about other, less-structured data, such as freeform text? Public web search engines have proven that people appreciate freeform query access across a variety of documents. You could build your own search engine, specialized for a given type of content or arena of knowledge, and see who will trade money for access. Many libraries are familiar with PsycINFO and ABI/INFORM, which are subject-specific search engines for psychology and business materials, respectively. Smiliarly, LexisNexis provides specialized search for attorneys and journalists.

Analyze

Analysis is another popular way to build a business around data. Given the media attention, one could argue that Analyze is even more well-known than its humble sibling, Collect/Supply. It's certainly been around a while—although terms such as Big Data and data science are relatively new, "traditional" business intelligence (BI) is an established practice inside many companies. In fact, while the initial usage of the term dates from 1958,[8] the modern usage has still been around since the 1990s.[9]

Whereas the opportunity in Collect/Supply is based on a straightforward transaction—trading money for data—earning money through data analysis is more of an indirect pursuit. There are three forms:

External
: Offer your services to analyze someone else's data

Internal
: Analyze your own data

In-between
: Analyze some data and sell the results

8. Luhn, H P. "A Business Intelligence System." *IBM Journal* 2, no. 4 (1958): 314.

9. "A Brief History of Decision Support Systems"

The unifying theme is to profit from insights that lie within a dataset.

External analysis involves providing data analysis services for third parties. On a rare occasion, someone will simply hand you a dataset and ask you to go panning for gold. More often than not, though, they'll start with a specific set of questions related to their business concerns: how to improve the bottom line (say, identify services they no longer use), how to improve the top line (find new ways to grow), or how to identify new markets (uncover new and unexpected uses for existing products). Certain sectors—notably mobile phone companies and retail services—have expressed interest in predicting customer behavior. While ad-hoc analysis is all the rage, there's still room for more practical pursuits such as audits and anomaly detection.

Such services are especially useful to companies that have data-related questions but don't have enough work to justify a full-time team of analysts. Note that you'll need domain knowledge to understand whether your findings are of any use to the client. It's entirely possible to find something that's novel but irrelevant. Similarly, you can "discover" something that is relevant but already well known in that field.

With internal analysis, your revenue opportunity is in making smarter, more informed business decisions. The prime example would be companies whose very business model is built on data, such as trading firms, which gather tremendous amounts of data to develop and drive their trading models. The idea also applies to companies that analyze customer data to build recommendation engines or to manage marketing efforts. On a smaller scale, this describes companies that measure internal activity to develop capacity-planning strategies. In all cases, the analysis is one step shy of the money itself: either it helps you cut costs (say, you identify fraudulent activity), or it helps you make more money (you identify a profitable, untapped market segment), but it never generates revenue unto itself.[10]

For the in-between case, you analyze data yourself and sell the results. For example, you could conduct a survey in anticipation that someone would buy the distilled information. Your value-add is that you took the trouble to conduct the survey and explore the data. This is the most difficult of the three Analyze variants because it requires you to opti-

10. One could argue that a recommendation engine generates revenue. We contend that a recommendation engine doesn't output *money,* it outputs *recommendations,* which may in turn lead to someone actually pulling out their credit card.

www.dbooks.org

mize across three axes: something that buyers would care about; something they don't already know; and something that matters enough for them to pay good money for it. You also suffer an additional risk in that you'll need to structure the sale in a way that discourages a secondary market from forming underneath you. You will otherwise sell your data only once, and that buyer will resell it many times over.

Because of its inherent risks, the in-between case works best as an extension of an existing business that already collects the necessary data as a byproduct of its operations. If you run a Collect/Supply effort and the data is very difficult for others to acquire, then such an Analysis effort can lead to additional revenues.

One caveat common to all three forms of Analyze is that this is a broad and open-ended topic. There are innumerable types of analyses one can perform. Even higher-level concepts such as "prediction" or "sentiment analysis" can quickly branch off into a variety of techniques. To build a business around Analyze, then, you would do well to develop a team in order to achieve the required breadth and depth of skills.

Obscure

Till now, we've explored strategies that involved bringing data in. There are also business opportunities in keeping data behind closed doors. For every firm that seeks to collect data, there are those who seek to hold their data private. Companies wish to protect their data in-transit so that others do not profit from data exhaust or similar byproducts. Individuals, as they learn more about the ways in which companies collect information about them, increasingly prefer to maintain their privacy. You can profit on this dichotomy by building tools to obscure information or otherwise foil data collection.

Marketers have been collecting contact info since the dawn of the database (or, perhaps since the dawn of the phone number), but it's clear that web browsing has opened up massive new opportunities for large-scale data collection and analysis. Web servers have been logging page requests for technical reasons since long before marketers knew to leverage the information. It's now possible for websites to transparently plant any number of cookies, web beacons, or other tracking mechanisms into a single web page. The result is that a casual browsing experience on one website can leave an activity trail through several

providers, who may in turn combine data from various sources to develop a profile of the unwitting end user.

This makes advertisers happy because they get access to detailed information that they can use internally or resell. It also discomforts a number of end users who would prefer to keep their browsing history a private matter. Browser plugins DoNotTrackMe (formerly DoNotTrack+) and Ghostery inhibit tracking. Both plugins are free, though there is certainly room for a paid service around this theme. Furthermore, Abine, maker of DoNotTrackMe, offers a paid service that promises to remove your information from data-collection sites.

Businesses are also concerned with privacy, and they go to great lengths to keep data inside the corporate walls through the use of VPN access and internal websites. That said, companies often have to exchange data with outside firms and may give away some of the shop in the process. Take Google as an example. End users' queries pass through a number of intermediaries on the way to the ubiquitous search box. ISPs and other network middlemen therefore have the opportunity to collect raw search queries, which can form a rich dataset. In May 2010, Google debuted a separate, SSL-encrypted version of its search page[11] and later enabled SSL on the main Google search page.[12] By plugging this data-transport leak, Google closed the door on a secondary market for its search query information, thereby improving its revenue opportunities: anyone who wants the search data must make a deal on the search giant's terms.

Another option would be to create a *service* around the theme of plugging data leaks, such as that offered by BrightTag. The company helps website owners limit the amount of data collected by partners and ad networks. Typically, you let an ad partner place their tracking tags on your website, which grants them unfettered access to your visitors' information. BrightTag's "smart tag" lets you collect all the information you want, but only pass along specific information (say, "user's geographic location") to third parties.

11. "Search more securely with encypted Google web search"

12. Granted, middlemen can see where you go if you click a Google search result link, but in your early stages of exploration, all they know is that you did some searching.

www.dbooks.org

Consult/Advise

Our last strategy, Consult/Advise, is the most open ended. Consulting is certainly not unique to the data arena, but it does come with its own playbook. Here, you make money by advising companies on how to use their data to get ahead. A consulting effort cross-cuts Analyze and, to a smaller extent, Obscure, so we won't rehash those ideas here. In some ways, though, the consulting in those examples—that is, the actual strategic guidance—is a smaller part of the overall effort. It's wise to consider cases in which the strategic guidance takes the main stage.

Consultants trade money for access to their experience and expertise. One key element of a data consulting operation, then, is domain knowledge beyond that of the client's industry.

For example, people with strong economics or finance experience can add a perspective beyond pure data analysis. They would readily see that airline frequent-flyer programs and retail points-for-participation services represent virtual currencies, with the airlines and retail firms acting as central banks. These firms risk the same issues as their real-world government counterparts, such as bank runs, sudden devaluation, foreign exchange (between real-world and virutal currency), and monetary policy. They have no doubt developed some interesting data by recording details of every transaction, but without skilled, specialized experience to help them understand that data, these companies risk virtual-economy problems that may leave them at a loss for real-world cash.

Consider video game company Valve, which hired an economist[13] in 2012 to assist with the virtual economies they were creating in their massive, multiplayer online games. We don't think Valve is rare because they engaged an economist, but because they openly acknowledged that they had done so. We have reason to believe that airlines, hotel chains, and other proprietors of virtual currency also take guidance from economists but have been rather quiet about it.

Consulting opportunities may also open up for data-minded people with experience in law enforcement or armed conflict. Social networks and mobile phone carriers have access to location and personal-interaction data, which puts them in a prime position to detect or predict large gatherings or even criminal activity. Approaching this

13. "It All Began with a Strange Email"

from a different perspective, a person with geolocation experience could provide consulting services to law enforcement or government agencies to help them identify potential for social unrest.

Also consider a person with a background in privacy law or public relations. Such an individual could help companies steer clear of data misuse that would lead to costly public backlash. For example, social networks Facebook, Twitter, LinkedIn, and Path have all been put under the spotlight for being less-than-clear about gathering end users' personal data from their mobile phones[14]

Note that having specialized, orthogonal domain knowledge is a key ingredient to a Consult/Advise effort, but it's hardly enough to stand on its own. The best consultants provide a mix of the domain knowledge *and* some technical skill to work with the data. This permits them to make sense of the data at a high level, and also to analyze it to support their findings.

Considerations

No idea here is a perfect moneymaker, nor is any particular idea categorically superior to any other. What constitutes a strategy's benefits and drawbacks will rely very much on your abilities and resources. To that end, we'll close the paper on some considerations that cross-cut the strategies we've presented: domain knowledge, technical skills, usage rights, and general business concerns.

We hope these will help you determine which strategies will work best for you. Note that you can rarely consider these in isolation; they overlap, even interleave, with the others. Mathematically speaking, you should treat this as an optimization exercise across multiple dimensions.

Domain Knowledge

Collect/Supply and Simplify Access require the least domain knowledge, other than having a hunch about what data would interest peo-

14. Each incident led to a number of articles and blog posts. What we present here is just a small sample. For a general overview, see "Your address book is mine: Many iPhone apps take your data", "The Facebook Scare That Wasn't" (Facebook), LinkedIn iOS App Grabs Names, Emails And Notes From Your Calendar" (LinkedIn), "Path iOS app uploads your entire address book to its servers" (Path), "Twitter stores full iPhone contact list for 18 months, after scan" (Twitter).

www.dbooks.org

ple. These have the lowest barrier to entry compared to the other strategies. You could even treat your offerings as a stock portfolio by hosting a variety of datasets not tied to any particular field of interest, with the expectation that one or two may eventually become wildly profitable.

By comparison, Store/Host requires a fair amount of domain knowledge if you plan to offer hosted analytics. You need to understand what analyses your customers would want, and you must understand the data well enough to provide the right numbers! Furthermore, understanding the sorts of answers your customers seek will help you plan your technical infrastructure. Having the right numbers won't matter if your system is too bogged down to provide them to your paying customers.

Domain knowledge is a core element of Enhance/Enrich, Filter/Refine, Analyze, and Consult/Advise. You're otherwise at a loss to remove records, correct values, or provide meaningful guidance. These strategies are best suited to those with work experience in the customer's domain.

Technical Skills

Technical skill—writing software or performing analysis—is required for all strategies, and is often a competitive advantage.

In most cases, you'll need to write custom software tools to acquire or process data. Knowing what's possible and reasonable in that realm will help you gauge the difficulty of the exercise before you start. Combining that with your domain knowledge, you'll be able to quickly determine whether, say, cleaning up a dataset for a Filter/Refine effort will be worth your while: "The source data is easy to acquire, but will require two weeks' effort to build the tools to process it, plus another several weeks' human interaction to filter out the bogus records. The target market is small, very picky about the data quality, and unlikely to pay well. Not worth it."

Compared to pure Collect/Supply, Filter/Refine will require automation, so you'll need someone to write the tools. If you already have a background in software development, you can keep your operation

lean by writing the software yourself.[15] At the least, software fits cases in which the work is dull, repetitive, and predictable. When it comes to data *interpretation* as opposed to pure labor—such as some text-mining exercises or spotting certain types of bad records—software sometimes pales in comparison to human interaction. Once again, having technical skills will help you decide when it's time to write software tools and when it's time to call in some interns. (Amazon's Mechanical Turk service can provide programmatic access to human eyes, perhaps the best of both worlds.)

Efforts built around Analyze require two forms of technical talent. First, data rarely arrives in a workable form, which makes data preparation a difficult, thankless, yet necessary precursor step to any analysis effort. Data prep can range from a robust, recurring job to a quick one-off for an initial exploratory exercise. The required skills vary accordingly. The former case will definitely benefit from a background in commercial or enterprise software. So will the latter, one-off case, though to a smaller extent. An analyst who writes particularly inefficient tools can cost you in terms of time-to-market and technical infrastructure. If those tools are fragile and break at inconvenient times, you'll also suffer customer defection.

Second, analysts will need a proper understanding of math, statistics, algorithms, and other related sciences in order to deliver meaningful results. They must pair that theoretical knowledge with a firm grasp of the modern-day tools that make the analyses possible. That means having an ability to express queries in terms of MapReduce or some other distributed system, an understanding of how to model data storage across different NoSQL-style systems, and familiarity with libraries that implement common algorithms.

Perhaps surprisingly, Obscure can require the deepest technical talent of all. You may have to write commercial-grade software to perform the obfuscation, so having a experience in such projects will certainly smooth the road. You may also need to analyze a client's technical stack to spot leaks and then propose solutions, so you'll certainly benefit from a broad range of hands-on IT skills as well as experience in business strategy.

15. Granted, you need to strike a balance between writing software and running the business, but in our experience, the tools are simple enough to write that one reasonably skilled person can handle both.

www.dbooks.org

Usage Rights

Compared to physical goods, data is a strange animal. It is intangible and easy to duplicate, which makes it equally easy to copy from someone else. Combined, these traits can lead to several issues around usage rights, and if you're building a business around data, rights issues can be a minefield.

First, there's the black-and-white issue of permissions. Before you collect the data for your Collect/Supply business, make sure that you're permitted to resell it. Data that is "free to see" may not be "free to use." Many websites are publicly accessible, yet commercial in nature, and their terms of service (TOS) expressly forbid scraping. You can still land in hot water if you manage to collect the data without being caught, because the data's source company may eventually find you and litigate you out of existence. Why start a business on such a risky premise?

Then, there's the grey area between the legal stance and the moral stance. Selling very fine-grained data, such as detailed personal information, can also get you into hot water. Even if your TOS covers the use and resell of the data, unexpected publicity around such use can quickly work against you. (Note the Facebook, LinkedIn, and Path stories we mentioned in the Consult/Advise section. Those companies survived, in part, because they had massive budgets to fuel their legal support and PR strategies. Unless you possess similar resources, you need to work harder to stay out of trouble.)

Business Concerns: Pricing Strategies, Economics, and Watching the Bottom Line

Many people developed an interest in data out of intellectual curiosity. That's a reasonable stance for someone doing the technical work around analytics, but if you're in charge of running the business, you need to focus on *turning profit.* Furthermore, if you're running a shop of just a few people—which is entirely possible, given modern technology—you may have to fill both roles at once, which means keeping them in balance.

Perhaps it goes without saying, but running a successful business venture based on these ideas will require conscious effort on your part:

To start, you'll want a healthy dose of forward-thinking to temper your rush to market. It's painfully easy to trivialize important matters that

don't provide immediate feedback. On the technical side, this can lead you to underestimate the costs you'll incur during design, development, or ongoing operations. On the business side, you could end up with a product that no one wants to buy. Worse yet, you could develop a wildly popular product that you later lose to a lawsuit because you weren't permitted to sell it in the first place.

It's equally important to gauge how your operations will scale. From your development partners, to your infrastructure, to your bandwidth, make sure you understand what will affect your costs. Perform sensitivity analysis around your estimates and explore different scenarios to see where trouble spots might arise. (If you've been knee-deep in the business side of things, this can provide an intellectual diversion.) While it's discouraging to see such obstacles during your planning stages, it can prove fatal to hit them unawares once you've established your business.

Create prototypes to explore cost estimates and scale, then run rigorous experiments and load tests. This will help you to identify nonlinear scaling issues, which can be especially damning if your service benefits from a sudden surge in publicity.[16]

Building on that last point, the old adage "it takes money to make money" still holds true in the data age. You'd do well to engage an experienced person to build out and tune your infrastructure and custom software. Wouldn't you rather pay a professional in money up front than pay in negative publicity after the fact? Simply put, you can't serve a customer if your site is offline.

For the strategies that involve a particular dataset—Collect/Supply, Enhance/Enrich, and Filter/Refine—make note of any special skills you have that would help you operate more quickly or more efficiently than your competitors. For example, say you build a platform to automate work that your competitors do by hand. That would let you woo clients with lower prices, or maintain higher margins.

Last, and certainly not least, we'd be remiss if we didn't explain that we barely scratched the surface on building a Consult/Advise effort. There is more to a consultancy besides domain knowledge and technical

16. This is sometimes known as the SlashDot effect, named for the popular website. A positive mention on SlashDot or similar sites can drive a lot of traffic your way in a short period of time, which can both boost your customer count and pummel your infrastructure.

www.dbooks.org

analysis skills. You also need to understand how to run a business, which includes everything from marketing to day-to-day operations. Most of all, it takes a certain personality to act as a professional outsider. Still, compared to the technology-driven strategies we've presented, a Consult/Advise business requires neither hardware infrastructure nor technical labor. That means it can involve a shorter time-to-market and stronger margins.

Conclusion

We hope this brief survey has given you some ideas on how to start or extend your data-based business. To begin, you could apply these strategies to your own data in search of ideas. You could also examine an established data business to better understand their value-adds and perhaps explore any avenues they seem to ignore.

Remember to treat this list as a general framework rather than a how-to manual. Furthermore, remember that the ideal business would build on a portfolio of strategies; resist the temptation to stop at the first idea that bears fruit. Once you establish your base and reputation, you should seek to leverage your data in as many ways as possible.

All in all, we wish you good fortune as you build your business in the realm of data.

About the Authors

Q Ethan McCallum works as a professional-services consultant, with a focus on analytics strategy. He is eager to help businesses improve their standing—in terms of reduced risk, increased profit, and smarter decisions—through practical applications of data and technology. His written work has appeared online and in print. He is currently working on his next book, *Making Analytics Work: Case by Case*.

Ken Gleason's technology career spans more than 20 years, including real-time trading system software architecture and development and retail financial services application design. He has spent the last 10 years in the data-driven field of electronic trading, where he has managed product development and high-frequency trading strategies. Ken holds an MBA from the University of Chicago Booth School of Business and a BS from Northwestern University.

www.dbooks.org