A DISCUSSION WITH
SAMEER SOLEJA.

Sameer Soleja founded Molecule Software in 2012 after a decade spent designing, building, and implementing systems for energy trading companies. Read on to learn more about the technology that powers Molecule, how it improves the productivity and quality of life of its customers, and how blockchain and the cloud are changing the world of energy trading.

MOLECULE
SOFTWARE

A DISCUSSION WITH
SAMEER SOLEJA.

APRIL 2017

PART 1

MOLECULE
BY
MOLECULE

Sameer, tell us your story. Where did you begin? What led you to start Molecule Software? Why did you think you could do it?

I spent about ten years at SunGard developing energy trading software. I went to business school, and when I came home, realized there had to be a better way than what we were doing. We were making and implementing multi-million dollar trading software that everybody hated, hated paying for, hated working on, and where implementations took a year-and-a-half of pain.

A great example was when I helped a gas company implement the industry's leading product, OpenLink Endur. They were paying up to $20 million to get it working—and during the months I was there, I saw a dozen people go home frustrated and angry every day.

To me, it seemed that technology had advanced to a point where this didn't have to be the case anymore.

We started Molecule with the idea that multi-million dollar technology should be more awesome than the 99-cent apps you use on your phone.

We have spent the last five years building and improving our technology. While our model has evolved, our core has stayed the same. Molecule is a SaaS app that provides way more value than it costs, and it helps people in the trading industry get their job done quickly and enjoyably so
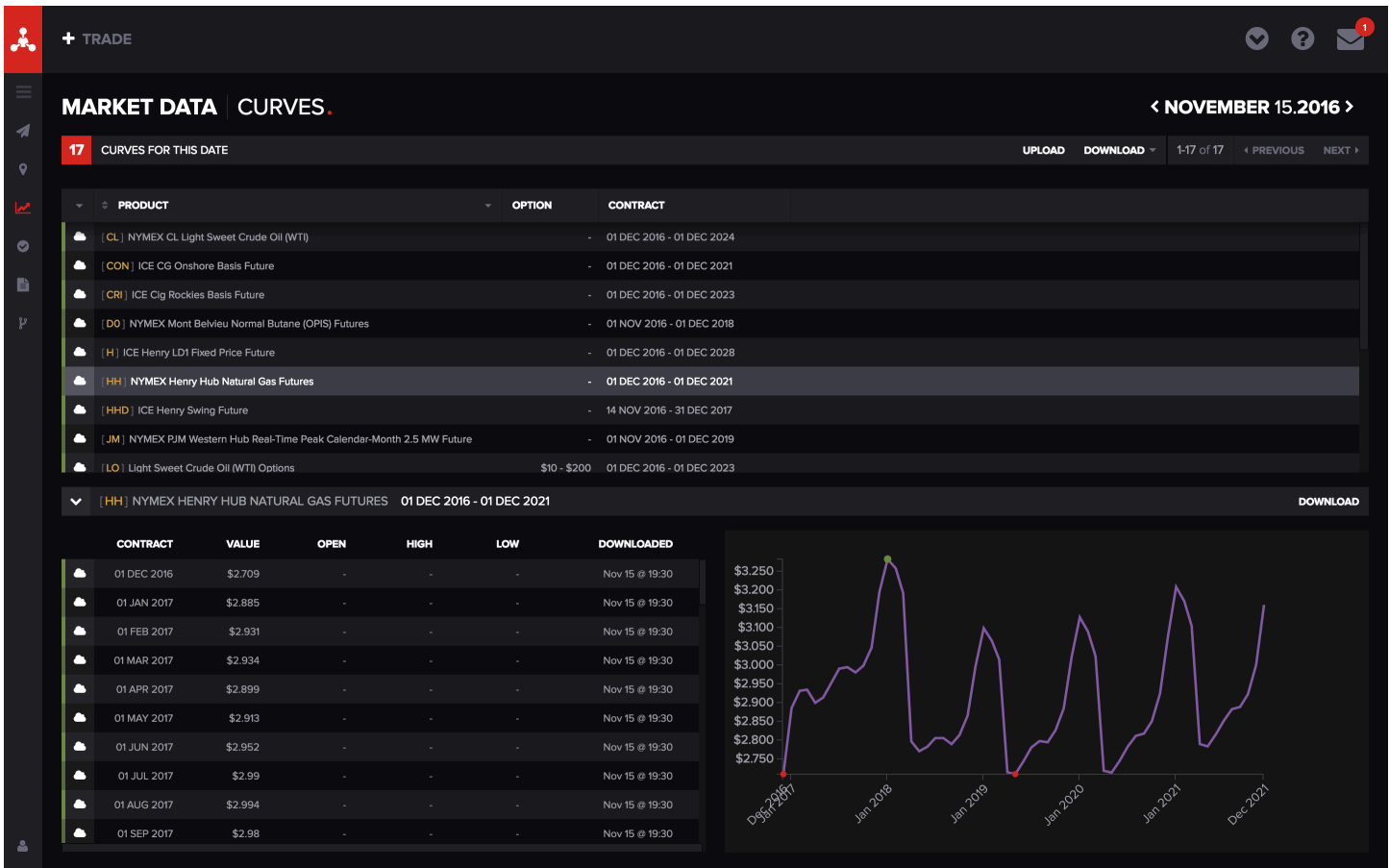
Molecule Interface

they go home early. Plus, our implementations don't take an army of consultants and millions of dollars.

What specific advancements in technology led you to believe that Molecule would be better than competitive products in the market?

One of the key things was the "app-ization" of software. When I grew up programming in the late 90s and early 2000s, nobody really cared about ease of use or design. Then around 2006, Apple came out with the first iPhone. We started to see the advent of purpose-built, relatively inexpensive, and really great software. At that time, ease of use had become a mantra in the consumer software industry, but it hadn't yet trickled to the enterprise. I saw an opportunity for that.

Secondly, to build an enterprise software company, you no longer need server farms. While you technically haven't needed server farms since the early 2000s, large software vendors were still deploying packaged software to companies, and I knew we could do better.

Thirdly, this is the age of open-source software. Open-source components have gotten so good that they are now the best in the industry. No longer do I have to pay five thousand dollars to Infragistics for a license to get a grid on my screen. Instead, I can go pick an open-source grid from GitHub, and it'll be way better than a packaged tool ever was.

The fourth piece of tech that has changed is big data, which has now sort of morphed into machine learning and AI. We saw big data gaining traction, and people were starting

to do interesting things with it like natural language processing. We thought that we could weave this into our software.

Thinking about it in terms of what a customer would want, would a customer pick an on-premise piece of software that looks like it is from their old 386 and runs on Citrix or something that costs the same price, but uses natural language processing to anticipate what they want to do and then just gets the job done so that they can go home early?

We thought we could be the latter and the available technology was right for it. Nobody else in the industry was doing it, so we thought the time was right for us.

Molecule also brings a different business model to the market. Can you talk about your service sales model and what this means for customers?

Our model is similar to any real software as a service (SaaS), and our pricing is pretty simple. Molecule costs one flat, monthly price for a package of users, a certain amount of fills per day, and the complexity of your portfolio.

For that price, you get it set up for you, you get it working creating P&Ls, and you get the custom reports and market data you need.

All without paying anything extra. There are no additional fees whatsoever.

We've had prospects be skeptical, because they're used to being surprised by other vendors who nickel-and-dime them out of things like market data. The norm in the industry is to quote a price, but then tack on about 75% of your total cost later (in consulting fees and add-ons).

We don't do that. Molecule contains everything you need to get going, and we'll get you up and running in 90 days or less.

This isn't rocket science. It's what every self-respecting software company outside the ETRM world does.

Our SaaS model provides benefits outside of pricing, as well. Molecule gets better every couple of weeks, with every new release. All of our customers get new features, bug fixes, and security updates at the same time, so you're never 19 versions behind and waiting for a vendor to do an upgrade for you.

In addition, this model also allows us to further assist operations groups by automatically analyzing trade data for oddities. The Molecule team can alert you when a trade doesn't mark or a P&L seems out of whack. Users can go

home at night confident that their reports are correct, because our whole team in Houston is watching out for them.

Molecule is not for everyone, is it? If you're a start-up prop shop with one or two traders and a spreadsheet system, you'll work well without forcing too many problems. Can you talk about the risks that arise at a four or five trader threshold and how Molecule solves those problems?

When you've just started a prop shop, and you've got one trader who has a specific strategy that they're following for themselves or one investor—you can, and probably should, manage that risk on a spreadsheet.

The problem with adding a significant number of traders is the same as adding a number of people in any organization. People do things in different ways, and they need to communicate with each other and know what the organization is doing as a whole.

That's what Molecule does. We pull the entire portfolio together so that operations people can see what's going on without having to clarify the intent of each trader in the organization. One of the risks that comes up when you grow past one or two traders is that

the traders and the principals may not know what the organization as a whole is doing. The number of investors or capital providers may have grown, and they may require a standardized set of reporting that doesn't involve humans, so that there is no possibility of human error or something more insidious than that.

Also, as the number of traders grows, so does the trade volume. What we've noticed in this industry is that the enterprise systems in a trade's path and lifecycle are not perfect. They're not like consumer banking systems— they tend to be much more primitive, and they make a lot of mistakes. Chasing down those mistakes all day starts to become an operations person's job, and that's just horrible. That's not what they want to be doing, and that's certainly not what you want to pay them to do.

Finally, as your trading organization grows, Molecule helps you enter new markets relatively quickly. We had a customer who decided all of a sudden that they needed to be exposed to a Chinese market. They informed Molecule after they had already deposited money at their FCM and made their first transaction. Molecule was able to support them, whereas their internal systems might not have done so for quite some time.

5

What advice would you give to a trading house that's weighing up the benefits of hiring a quant to build an in-house model to do what Molecule does, versus licensing Molecule?

Your lifetime cost of building an in-house model will be far higher than what you pay Molecule. We run into that all the time—for instance, when funds have hired a quant to build out a set of systems for them. While a quant can do that and cobble together something that works, we generally find

we use the same code for everybody. We also have our own set of controls that regression-test the heck out of our software multiple times a day. So, you don't have to worry that your decisions are informed by bad math. Your reports are right.

That's an interesting point that you've made about quants not being software developers, Sameer. Can you talk about the range of disciplines that you brought into the team to build Molecule? What are the various skill sets that your team has?

> **"You probably won't find a team anywhere else that has that sort of experience blended together to give our customers easy-to-use, bulletproof software."**

that quants are not professional software developers. They're very good at what they do, which is really hard math.

But building robust systems that are developed quickly, reliably report numbers, stay online without errors, and scale with your business, is what a software company does—and with Molecule, these things are exactly what you get.

A great example is a fund that, before bringing on Molecule, was using an in-house, quant-built system. As part of our onboarding process, we attempted to have Molecule's results match their pre-Molecule numbers.

We could never get certain numbers to match, and after quite some time trying to figure out why, we realized the problem: the quant had made a mistake in how they calculated option Greeks. In turn, the client's option Greeks were wrong—5-10% for first-order Greeks, and > 100% for second-order ones.

Molecule takes away the concern that a quant might have made a one-off mistake, because

Absolutely. Our team not only includes experts in usability design and the finance and energy industries, but our software development team comes from a variety of backgrounds that include banking, financial services, exchange order routing, and hosting. All of this expertise and experience comes together in our product to ensure that we have software that is professionally written for scale, performance, and is monitored so that, if anything fails, it's immediately fixed. Finally, we've enlisted the help of a number of quants to help build the more analytic parts of our system, which include our Monte Carlo Value at Risk (VaR).

Molecule sits at the intersection of bleeding-edge tech, finance, the commodities industry, operations, and design. You probably won't find a team anywhere else that has that sort of experience blended together to give our customers easy-to-use, bullet-proof software.

PART 2

BUILDING A STELLAR RISK SYSTEM

Pivoting to what's happening with the product in 2017, you've recently invested in new business intelligence software to enhance Molecule. Can you tell us how this has improved your model and what feedback you have received from customers?

We have been trying to deliver a consistent user experience to people across the commodities industry. Conventional wisdom has been that this can't be done because everybody wants something different. Embedding a BI tool is how we got around that. What we found is that 80 to 90 percent of the time, what people in the commodities trading industry want is the same.

But everybody seems to have some odd requirement or a different way they'd like to see a report. So we baked a reporting tool directly into our product to give our customers the personalization they needed. This was so successful that our customers started asking for more and more reports—even ones that worked like miniature apps.

In early 2017, we baked in a BI solution called Mode Analytics to replace our old reporting tool. This tool enables us to report on literally any piece of data that we store for customers. It has gorgeous custom visualization and a variety of charts as well as tabular data, and it enables us to use straight SQL and Python to shape the customer's data.

Now we're able to support more rapid development of these reports. In addition, reports are far easier to use and can be much more complex than what we had before.

Molecule Reporting
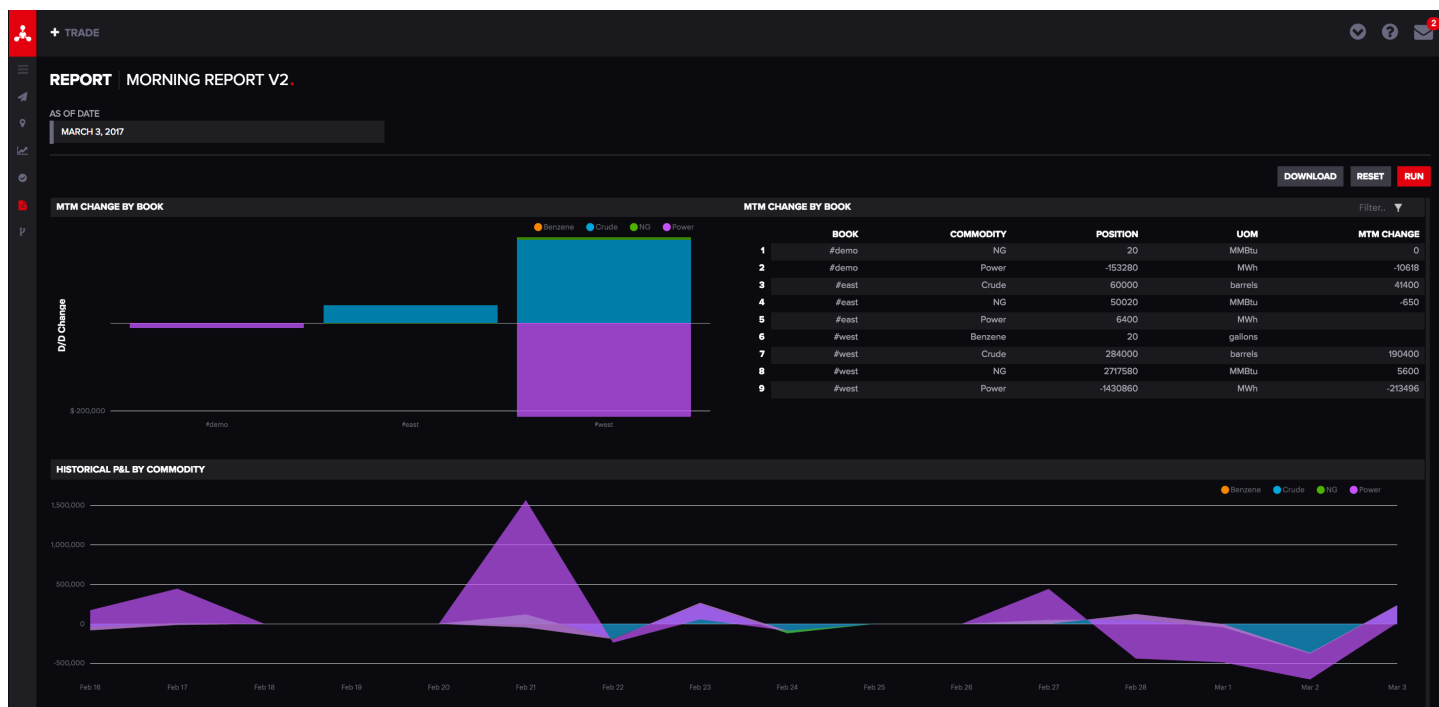
A great example is from a customer who trades a lot of power, who asked us to frame our megawatt-hour numbers in megawatt terms. At the time, we didn't have that capability, but we were able to roll it out in a report for them within days.

There's been a lot of work in recent months to improve usability, functionality, and risk analysis. Can you tell us about some of those enhancements and what they might mean for the users of Molecule?

Other than new features, one of our biggest internal initiatives this year is a six-sigma type quality focus. Internally, we are tracking every customer report of an issue and doing a root cause analysis. The idea is to keep doing these "andon cord" pulls, so that we build monitoring and processes that prevent the issue from ever happening again.

Our goal is to be just as reliable as any other cloud service you use (even at home), and we've done a lot of things to pursue that.

One of the things I touched on earlier is

automated risk analysis by our team in Houston. We're also rolling out a suite of tools to help customers diagnose problems more easily and to give them a better understanding of what's happening under the hood, so they can resolve a problem themselves within seconds instead of having to contact our support team.

Finally, we've been working on scalability. We've beefed up the underlying hardware for our environments and now have almost double the capacity we did before. We have also introduced auto-scaling to our environments, so that additional capacity comes online exactly when it's needed.

While developing these risk analysis tools and monitoring systems, what has surprised you about risks that companies using less robust systems, such as spreadsheets, are exposed to on a daily basis?

The biggest surprise is just how hard operations people's jobs are. We did an exercise where we mapped out an actual day in the life of one of our customers, and we

realized that our customers have really, really hard jobs.

We focus on everything we can do to make their lives easier and to enable them to not have to worry about making a small arithmetic error somewhere in the reams of data they have to process every day. From a risk perspective, we have realized just how much our customers have to process and how much room there is for "fat-finger" error. Once again, Molecule is continuing to try to eliminate human faults.

Do you have any horror stories of things going pear-shaped when an in-house system failed and how Molecule has been a safety net for a company?

One of our clients is a hedge fund that decided to build their own ETRM in-house.

They had decided, and rightfully so, that computing a mark-to-market is really easy, so How Hard could this Whole Thing Be? And they're right! Computing a mark-to-market is really easy: current minus prior times quantity.

But they ran headlong into the legacy aspect of the rest of the industry. They built an exchange connector to download their trades automatically from an exchange and fought with the thing for six months. They couldn't figure out why they were getting duplicate trades every day or why they weren't getting some trades. Finally, one of the fund principals decided they had poured enough money into it and that they were just going to go back to spreadsheets.

And they did, for ten years, before finally encountering Molecule. We had them up and running in about a month. Today, all of their users use Molecule except for the ones who

Molecule Reconciliaction

couldn't be pried from their spreadsheets. Those users instead have spreadsheets connected to Molecule's Excel APIs that look exactly like their old spreadsheets, but are more reliable than their old system could ever have been.

We've previously spoken about Molecule's VaR being a game changer. Can you give us an update on how Molecule's VaR looks now and what progress has been made in the last six to eight months?

One of the things that seems to be unique to VaR in the trading space is that consistent opinions on how it should work, what it should do, and even what math it should use, are hard to find. It's a heavily statistics-driven calculation, and what we find is that most customers have a high-level understanding of how it works, why it works, and what it does, but everybody has a different opinion on the details.

We spent the last year or so validating our Monte Carlo VaR calculations against not only every textbook we could find, but also our customers' portfolios and what they expect their VaR numbers to be. We want this number to be as worry-free as anything else in our software, in addition to it being the fastest and most automatic VaR available. We're nearing the end of that process, and when our VaR module comes out of beta in the next few months, our customers can be confident that they have a VaR number that just works. At that point, the days of standalone Monte Carlo at Risk systems are numbered.

Do you feel that the work that has been done by Molecule in recent months is unique in the industry?

We feel that our offering is unique in that an accurate VaR number can be had without a customer ever having touched their portfolio. We have customers who don't trust the VaR numbers from their ETRM systems

because they don't take the peculiarities of their portfolios into account and just report gobbledygook instead. Our customers end up manually exporting data from an ETRM system into a VaR-specific or a VaR-focused system to get a number that they remotely trust. So we think that our offering—in that it's integrated and doesn't have to be adjusted for the numbers to be trustworthy—is unique in the ETRM market.

What other features will you be rolling out in 2017?

One big suite of features that we're focused on this year is self-administration—to allow our customers to administer their account, add users, and tweak settings for things like their VaR without needing to call support.

The other, and larger, set of features is around physical scheduling. Plenty of our existing customers and folks we've talked to in the market need a CTRM solution that handles both physical and financial commodities. To date, Molecule hasn't had a physical scheduling piece. We are getting ready to roll the first beta of our power scheduling module. It handles US power, and can be expanded to handle European power as well.

Later this summer, we will roll out the first beta of our pipe and bulk scheduling. We're taking a very light approach to these features—modeling our users' primary needs first, then automating where applicable.

We intend to have v1 of our multi-modal scheduling functionality generally available by the end of 2017.

PART 3

TRADING TRENDS IN 2017

What are the trends that you see emerging in 2017? How will they affect traders and the way they do their jobs?

We're starting to see increased interest in financial trading on platforms other than the two big US exchanges. This is a great use case for Molecule, because processing trades and market data from five exchanges in three currencies (something very cumbersome to do by hand) is what Molecule does without even blinking.

We're also seeing an increased amount of interest for physical trading in the market, such as with multinational traders and banks.

Also, the technology market is continuing its steady march through machine learning to AI and commercially available AI solutions. There are a lot of interesting things going on in the market, some especially with IBM's Watson, and we are monitoring that space closely. That's an area where we can play really well and potentially provide predictive analytics to our customers.

Finally, blockchain. There is something to blockchain in how it is used in financial trading. At its most extreme, it could disaggregate trading from the exchanges themselves, and we've seen exchanges as well as large financial institutions start to embrace it. It will be interesting to see where it goes, and we will be prepared to help our customers use it.

A lot of research in the market is talking about cloud technology—what's the latest on cloud adoption in E/CTRM?

There was a great article in Risk Magazine a couple of weeks ago about that very thing. CTRM buyers are no longer as afraid of cloud technology as they were a few years ago, and even the supermajors are looking at how they're going to adopt the cloud to reduce their capex. Yet, we've still seen relatively low uptake of cloud solutions in the market, and we think there are a couple of reasons for that.

Most of the cloud solutions that are available are not real cloud solutions. They are an existing vendor throwing an install of their software onto a server that happens to be located within Microsoft's data center, changing the pricing model by splitting it into a monthly fee, and calling it a cloud solution. Customers don't realize any of the real benefits of a cloud solution, from continuous upgrades to ease of use.

Also are only two cloud vendors in this market, and the common complaint that we've heard is that cloud solutions for CTRM don't offer enough features. Basically, while we're seeing interest in the market, the two major cloud solutions - including us - don't offer the features that everybody wants. We're aiming to rectify that by the end of this year.