CREATING PACKAGED IP FOR BUSINESS ANALYTICS PROJECTS

A PERSPECTIVE FOR SYSTEMS INTEGRATORS

Sponsored by Microsoft Corporation

David Chappell
& Associates

Systems integrators with business analytics practices—business intelligence, data warehousing, and more—typically offer a mix of pure custom development and services built on pre-packaged software. Going forward, SIs should adjust this mix to provide more software, i.e., more packaged IP. Doing this will let you differentiate your business, improve margins, and win more deals.

Don't be put off by the word "software". While most packaged IP is provided as software, moving in this direction doesn't mean that you're becoming a software vendor. But the unavoidable reality is that changes in technology, such as the rise of cloud computing and the increasing availability of public data, are bringing changes to your business. To maintain profitability, you also need to change.

WHAT IS PACKAGED IP? CATEGORIZING THE OPTIONS

How much pre-existing software should you bring to a new engagement? There's no one right answer—it depends on the project.

As **Figure 1** shows, there are several options on the range from pure service, where an SI provides no packaged IP, to pure product, where an SI provides nothing but packaged IP. Each option on this line has value in the right situation, and so each is worth considering.
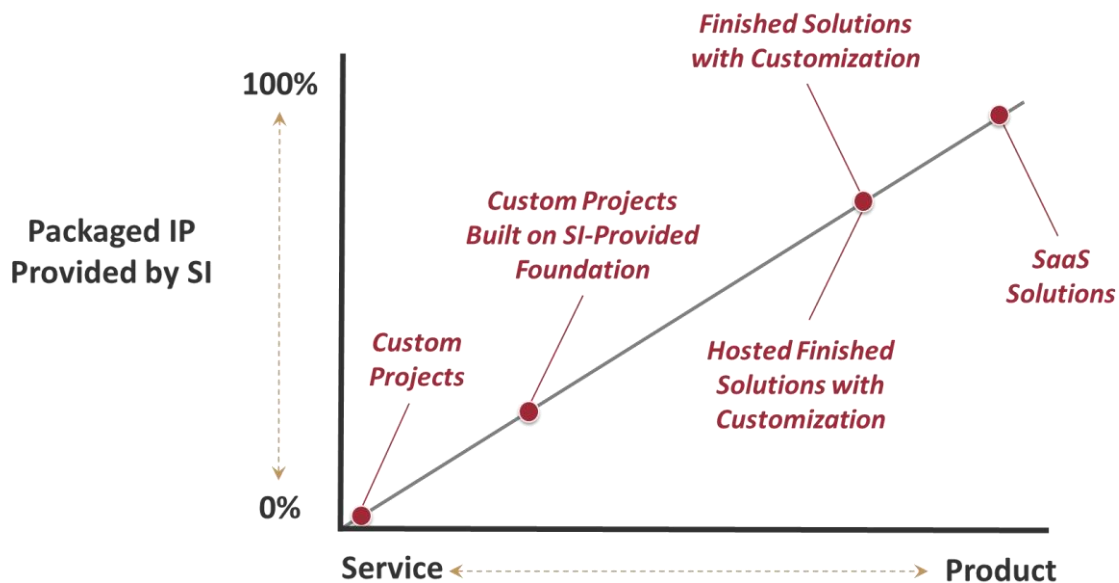
Figure 1

An SI can provide packaged IP at various points on the range between service and product.

CUSTOM PROJECTS

These projects use no pre-existing software at all. Everything is built from scratch for a specific customer, and so the SI's revenue comes entirely from services.

CUSTOM PROJECTS BUILT ON AN SI-PROVIDED FOUNDATION

Since SIs frequently do similar work for different customers, it's often possible to create libraries or frameworks that can be reused across multiple engagements. The SI doesn't charge separately for these solution accelerators, however, so once again, its revenue comes entirely from services.

FINISHED SOLUTIONS WITH CUSTOMIZATION

If an SI can solve very similar problems for multiple customers, creating a finished solution—a product—can make sense. This software solves most of the customer's problem, with the SI providing customization services for the rest. In this case, the SI charges for both the product and the services. For example, 60% of a project's revenue might come from selling the product, with 40% from customization services. Over time, this ratio might shift, with an increased share of your revenue coming from selling the product.

HOSTED FINISHED SOLUTIONS WITH CUSTOMIZATION

As in the previous option, the SI solves the customer's problem by selling finished software and customization services. In this case, however, the SI also runs the solution for the customer. The customer might pay directly for the product and customization, then pay a monthly fee to the SI for hosting and managing the application. Alternatively, the SI might charge nothing up front, but instead sell the hosted solution back to the customer at a higher monthly fee.

SOFTWARE AS A SERVICE (SAAS) SOLUTIONS

If many customers need the same solution, an SI can create a finished application. Today, that application is likely to run in the cloud as a SaaS solution, with the SI selling per-user per-month subscriptions. As Figure 1 shows, a SaaS solution is a product rather than a service. An SI that provides this option is now both an SI and a cloud-based independent software vendor (ISV).

WHY OFFER PACKAGED IP? THE BENEFITS FOR SYSTEMS INTEGRATORS

Being a systems integrator means providing services. But adding more packaged IP to your portfolio can offer real benefits.

Whatever option you choose, creating packaged IP isn't free. You need to fund the creation of frameworks and solutions from the cash flow generated by your services business. You also need to be careful not to let the immediate attraction of services revenue tempt you to take skilled people off internal development projects. But if it's done right, creating and selling packaged IP can significantly improve your business. Here's how.

YOU CAN EARN HIGHER MARGINS THAN WITH PURE SERVICES ENGAGEMENTS

Like it or not, development skills are becoming a commodity, even in more specialized areas such as business

intelligence and data warehousing. The lower hourly rates this implies shrink your margins, making your current business model less attractive. With packaged IP, however, more of your revenue comes from selling reusable software. You can now price based on business value rather than solely on billable hours. Doing this successfully is likely to result in higher margins and a higher valuation for your firm.

YOU CAN DIFFERENTIATE YOUR BUSINESS

In most business analytics markets, differentiating your firm is hard—many SIs provide similar services. If you create a framework for a particular area, however, or build a more complete solution, you demonstrate your knowledge of and commitment to this market. Packaged IP gives you a clear differentiator from your competitors.

For smaller SIs, offering a better solution for specific problems can also help win business that would otherwise go only to global SIs. While you might not be as big, your investment in packaged IP can let you provide a clearly better offering in some areas.

YOU CAN OFFER FASTER PROJECT COMPLETION TIMES

With packaged IP, some of the work required for a project is already done. This lets you complete projects in less time, giving you an edge when you're competing for work. Especially as the buyers for business analytics projects shift from IT leaders to business leaders, providing fast results becomes increasingly important.

YOU CAN LOWER THE RISK OF PROJECT FAILURE

You know that your clients will pay more to lower their risk. Packaged IP helps do this by letting you start a project with proven solutions for at least part of the required work. And as with faster completion times, tangible evidence of lowered risk—existing software in a particular area—can help you win more deals.

YOU CAN MITIGATE THE EFFECT OF SKILLS SHORTAGES IN SOME AREAS

Some hot business analytics technologies, such as Hadoop, require specialized skills that aren't yet widely available. Creating packaged IP that addresses specific problems here, such as sentiment analysis, can let your firm sell more projects in these resource-constrained markets.

YOU CAN REPLACE REVENUE LOST TO CLOUD COMPUTING

In a typical project today, your firm probably spends a chunk of time setting up infrastructure: installing business analytics software, configuring that software, and more. In the era of cloud computing, however, much of this work will already be done.

For example, public clouds today commonly provide virtual machines on demand with business analytics software already installed. Even the private clouds used by some organizations offer similar efficiencies. While this can shorten the time needed to complete a project, it also reduces your billable hours. Providing more packaged IP—which lets you offer more business value—can help you make up for this revenue loss.

CREATING PACKAGED IP: THE MICROSOFT BUSINESS ANALYTICS PLATFORM AS A FOUNDATION

Creating packaged IP means choosing a platform to build on. Porting your software across multiple vendors is likely to be prohibitively expensive, and so choosing the right foundation for your business is a big decision. One option is the Microsoft business analytics platform.

Describing the Microsoft Platform

As **Figure 2** shows, the Microsoft business analytics platform includes a number of technologies in several different areas.

DATA INTEGRATION

SQL Server Integration Services (SSIS), provided as part of **SQL Server**, supports extract/transform/load (ETL) projects for creating data warehouses and more. For self-service integration scenarios, the platform provides **Power Query**, an integration add-in for **Excel**. Power Query allows access to both internal and external data sources.
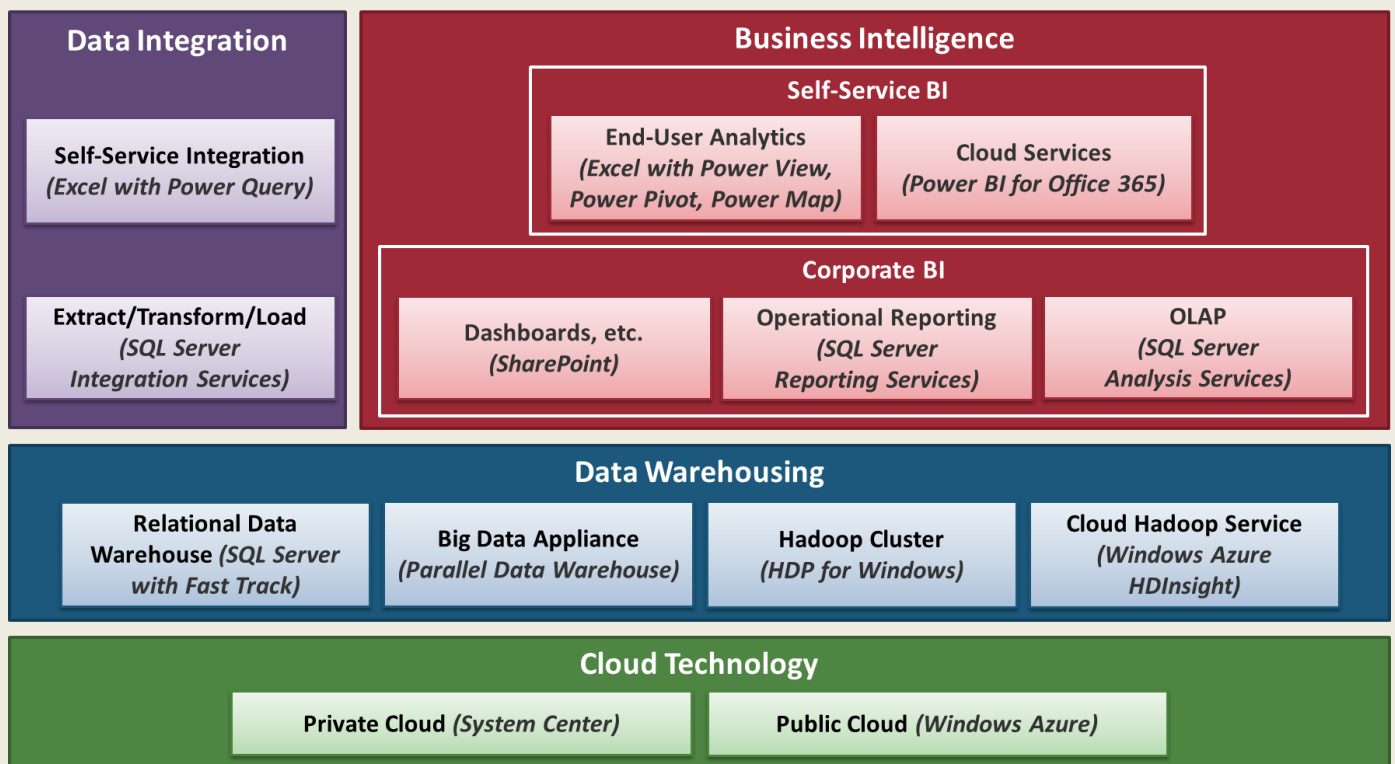
Data Integration	Business Intelligence		
	Self-Service BI		
Self-Service Integration *(Excel with Power Query)*	**End-User Analytics** *(Excel with Power View, Power Pivot, Power Map)*	**Cloud Services** *(Power BI for Office 365)*	
	Corporate BI		
Extract/Transform/Load *(SQL Server Integration Services)*	**Dashboards, etc.** *(SharePoint)*	**Operational Reporting** *(SQL Server Reporting Services)*	**OLAP** *(SQL Server Analysis Services)*

Data Warehousing

Relational Data Warehouse *(SQL Server with Fast Track)*	**Big Data Appliance** *(Parallel Data Warehouse)*	**Hadoop Cluster** *(HDP for Windows)*	**Cloud Hadoop Service** *(Windows Azure HDInsight)*

Cloud Technology

Private Cloud *(System Center)*	**Public Cloud** *(Windows Azure)*

Figure 2

The Microsoft business analytics platform includes a variety of technologies.

DATA WAREHOUSING

SIs can use **SQL Server** to create relational data warehouses and data marts. Microsoft also provides **Fast Track** architectures, which specify predefined hardware and software configurations for various warehouse scenarios. For handling large amounts of data, the platform includes **Parallel Data Warehouse (PDW)**. This big-data appliance supports up to five petabytes of relational and non-relational data. And for large amounts of non-relational data, the Microsoft business analytics platform includes two more options: **Hortonworks Data Platform (HDP) for Windows**, a Hadoop offering, and **Windows Azure HDInsight**, which provides a managed Hadoop cluster in the public cloud.

BUSINESS INTELLIGENCE

As part of SQL Server, the Microsoft business analytics platform provides **SQL Server Reporting Services (SSRS)** for operational reporting and **SQL Server Analysis Services (SSAS)** for OLAP. For creating dashboards, applications built on this platform rely on SharePoint. Microsoft also provides support for self-service BI using **Excel** with **Power Pivot**, **Power View**, and **Power Map**. To extend this to the cloud, the platform includes **Power BI for Office 365**.

CLOUD TECHNOLOGY

Both private and public clouds are important for business analytics. Microsoft lets organizations create private clouds with **System Center** and also provides **Windows Azure**, a public cloud platform.

Why Build on the Microsoft Business Analytics Platform?

Many vendors provide platforms in this area. Here are some of the reasons that SIs have chosen to create packaged IP on the Microsoft business analytics platform.

MICROSOFT PROVIDES A BROAD PLATFORM FOR PACKAGED IP

This breadth lets you create solutions that span diverse areas. For example, a data warehousing framework might combine PDW with HDP for Windows to support large amounts of relational and non-relational data. Similarly, self-service BI solutions can use Power Query to access SQL Server data marts created for the corporate BI world. Microsoft also provides Windows Azure, which SIs can use to host finished solutions or run complete SaaS applications.

MICROSOFT RELIES ON PARTNERS FOR PROJECTS

Many software vendors in this market also sell custom development services. An SI building on technologies from these vendors might find itself competing with the vendor's own service offerings. Microsoft is focused on software; it doesn't compete with its SI partners for services projects.

MICROSOFT'S BROAD PLATFORM CAN HELP YOU SELL NEW PROJECTS

Because the Microsoft platform probably addresses a large part of your customer's business analytics requirements, you can use an initial project as a beachhead to sell more work in other areas. For example, you might use an ETL framework to help win an engagement for creating a data warehouse,

then sell a self-service BI project to another client in this same company.

MICROSOFT'S SOFTWARE IS COMMONLY LESS EXPENSIVE THAN THE ALTERNATIVES

This can help you close deals, since the total project cost will be lower. It might also leave more money in the customer's budget for your services.

MICROSOFT'S SALES FORCE CAN BRING YOU CLIENTS YOU MIGHT NOT HAVE FOUND ON YOUR OWN

The Microsoft business analytics platform is a foundation for solutions; it's not really a solution by itself. Yet customers want solutions, not just platforms, and so Microsoft's salespeople are always looking for packaged IP that meets customer needs. Once they're aware of what you can offer, they'll often bring your firm in if they think it will help them close a deal.

EXAMINING PACKAGED IP: BUSINESS ANALYTICS EXAMPLES

To get a more concrete sense of the possibilities, it's useful to look at a few examples in each of the categories defined earlier.

Custom Projects Built on an SI-Provided Foundation

Many business analytics projects implement essentially the same process. Each customer has its own unique requirements, though, so the process commonly varies in some ways. In situations like this, creating a framework to support the process can be a good idea.

Think about ETL, for example. This kind of work might well be responsible for a large share of your consulting revenue—it's a critical part of many business analytics projects—and there are strong similarities across customers. Rather than recreate this wheel for each engagement, you might choose to build a framework that you can bring to each project.

Exactly what this framework does can vary with the kinds of projects you commonly do. Perhaps you'd benefit from standard SSIS code to create dimension and fact tables, for instance, or a metadata layer for SSRS, or something else.

Whatever the situation, bringing this kind of packaged IP into a project has many benefits:

- You can finish the project faster, since a significant share of the work is already done.

- You can ensure that your firm's best practices are followed, since they're baked into the packaged IP of the framework. This might include practices for quality assurance, collecting metrics, and more.

- You can more easily move your people across different projects. Since your developers already know your standard framework, they need learn only the business specific of a new project. This helps you deploy your resources more intelligently.

Frameworks aren't products—they're given away as part of a services engagement—and so the support requirements aren't the

same as with products. In some cases, though, it might be worthwhile to create process guidance for your client on how to use your packaged IP. For example, providing white papers as part of your packaged IP can help ensure the project's long-term success.

For some kinds of business analytics projects, a framework is all but essential today. There are many markets in which selling an ETL project without a framework puts an SI at an extreme disadvantage—its competitors all have one. There are other areas, though, where creating the right framework can provide unique differentiating value for your firm.

Finished Solutions with Customization

Frameworks make sense when you have multiple clients implementing similar but variable processes. But what if those variations aren't very large? If the majority of what your customers need doesn't change, you're probably better off creating a finished solution.

Finished solutions are essentially products, and they're probably used directly by your customer. Because of this, creating a finished solution is significantly more work than creating a framework that's used primarily by your own developers. Given the amount of effort involved, and the value the software provides, it makes sense to charge for finished solutions. Unlike frameworks, which are provided for free as part of a services engagement, your customers need to explicitly pay for a finished solution.

There are many possible examples of finished solutions for business analytics projects. They include these:

- **Sentiment analysis:** Every marketing professional—CMOs, brand managers, and others—wants to understand what their customers are saying about their offerings. Given the volume of information in Facebook and Twitter, this is a big data problem that can benefit from a Hadoop solution. A finished solution that helps them, say, measure the impact of a new marketing campaign might use HDP for Windows to process data from social media. For each customer, the SI might then do custom integration that combines these results with corporate sales figures gleaned from self-service BI information. Many customers want this, which makes sentiment analysis an attractive choice for finished solutions.

- **ERP surround:** Traditional ERP software often has limited support for modern reporting. A number of customers can benefit from stronger services here, which makes this another attractive area for finished solutions. For example, an SI might provide software that rebuilds an SAP cube in SQL Server, then creates modern SSRS reports on top of this cube. Another option might be to create a packaged self-service BI solution for this scenario using Excel and Power BI for Office 365.

- **Human resource analytics:** Just as marketing leaders want to know what customers are saying about their products, a company's leaders want to know what their employees are saying about their own organization. A finished solution for HR analytics might use HDP for Windows or Windows Azure HDInsight to analyze large amounts of non-relational data: training comments,

texts, Yammer posts, and more. This is essentially sentiment analysis for the enterprise, and a well-designed solution can give business leaders significantly greater insight into what's happening inside their organization.

- **Applications that provide access to external data**: One of the biggest changes happening in business analytics today is the increasing availability of external data. Whether it's demographic data, business data, weather data, or something else, packaged IP can help customers make better decisions. For example, think of an application that combines weather data with demographic data for a particular region, then analyzes this combination to predict sales of certain items when a storm is on the way. A variety of retailers might find this useful, perhaps with customizations that use each retailer's internal data. This solution might run on Windows Azure, on a private cloud, or on some combination of the two.

Even with finished solutions, an SI is still selling some services. Finished solutions make sense when a reasonable number of customers have common needs, with the differences addressed through customization services.

One challenge with this approach, especially for smaller SIs, is that customers might resist buying packaged IP from a non-traditional software vendor. The customer might insist on escrow of the code, for example, or other guarantees. In some cases, this concern can be mitigated if the customer is buying an external service rather than a product that they own and

run themselves. The next category of packaged IP makes this possible.

Hosted Finished Solutions with Customization

Suppose you have several potential projects that want very similar things, so creating a finished solution makes sense. But what if those customers don't want to run the solution themselves? As organizations increasingly move to the public cloud, this scenario is becoming more and more common.

For situations like this, an SI can both create a finished solution and take on the responsibility for running it. These solutions aren't shared—each customer has its own customized instance—but the core software is the same.

Many of the examples in the previous section could be implemented in this way. There are some situations where a hosted approach might be especially attractive, however.

For example, as **Figure 3** shows, a sentiment analysis application might be implemented using Windows Azure HDInsight rather than the on-premises HDP for Windows. This would make it easier to resell the solution to multiple customers.

Another example is the category of problems mentioned earlier that require access to external data: weather data, business information from Dun and Bradstreet, demographic data, and more. There's more external data available every day, and so helping customers take advantage of it can provide real value.

For SIs, creating a hosted solution lets you completely control the infrastructure the solution runs on. Rather than building

software that must perform well on different hardware configurations at each customer, the solution can be designed for just one foundation.

Another important difference between a hosted finished solution and a finished solution run by the customer is the revenue model. With hosted solutions, the SI commonly charges a monthly fee. If the SI has also charged for development services, this fee covers the cost of running the service (with a markup).
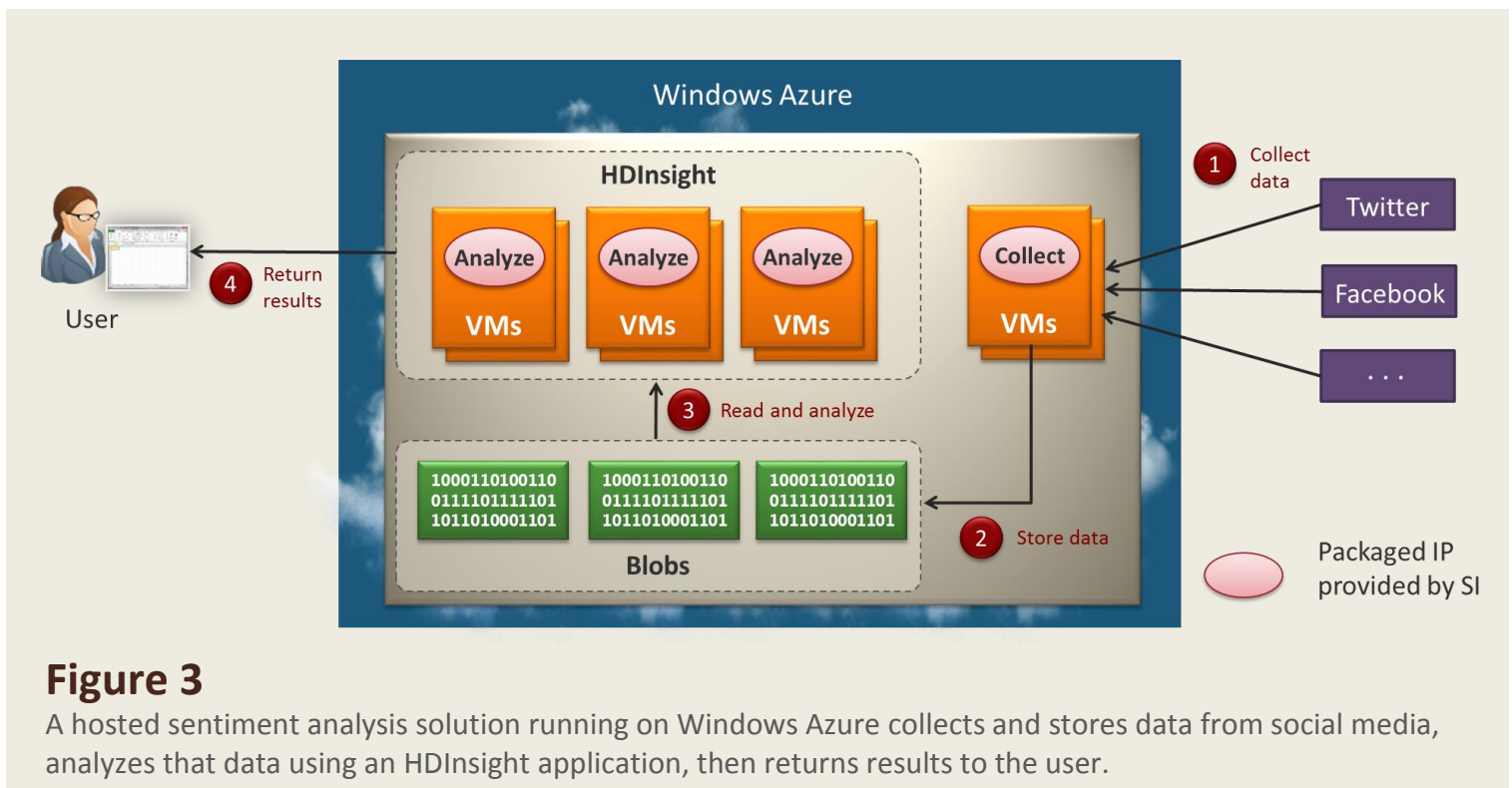
Some SIs, however, are using this approach to create what are essentially single-user SaaS applications. Rather than charging an up-front development fee, the SI gets all of its revenue from monthly charges paid by each customer. This can be riskier (even if the customer has a required minimum subscription), but it can also be more lucrative in the long run.

SaaS Solutions

Suppose you identify a customer need that's common across many organizations and requires minimal (or no) customization. Your firm could create a finished solution, then try to sell a (possibly hosted) copy to each customer. Or you could simply build a product: a SaaS application.

For example, suppose you create a Windows Azure application that collects and analyzes song performance data from YouTube and other internet sources. Your application could do this using SQL Server and SSAS running on Windows Azure, then provide a browser-based service to users. You might also expose this information for consumption through self-service BI with Excel. In either case, providing this service as a SaaS application might well be the best approach.

Unlike a hosted finished solution, a SaaS application is multi-tenant. In other words,

Figure 3

A hosted sentiment analysis solution running on Windows Azure collects and stores data from social media, analyzes that data using an HDInsight application, then returns results to the user.

all of your customers are essentially sharing the same code. While there might be opportunities for configuration, using shared code limits how much the application can be customized for each customer. These limits bring big benefits to you, however, since they also significantly lower the price of running and updating the application.

More and more SIs are creating SaaS applications today, driven largely by the search for higher margins. But don't underestimate the changes this requires to your business model. The business aspects of a SaaS application are different in almost every way from selling services engagements. Here are some examples of what changes:

- **Target customers:** Rather than selling projects to business and IT leaders, SaaS applications are commonly purchased (at least initially) by end users, the people who actually need what the application provides.

- **Channels:** Web-based solutions, especially at lower price points, commonly rely largely on web-based sales channels rather than relationship selling.

- **Revenue:** Customers commonly expect to buy SaaS applications through per-user per month subscriptions. While this can bring plenty of profit over time, the money takes longer to come in than with a services engagement.

- **Cost structure:** Providing a SaaS application obligates you to pay for running the software. This also means

that your cloud platform provider, such as Microsoft with Windows Azure, becomes a key partner.

If you find an opportunity to provide a large-scale repeatable offering with minimal customization, don't be afraid to create a SaaS offering. Be aware, though, that you're now entering a quite different business from the one you're in now.

SUMMARY

If your customer needs you to solve a very specific problem, that's a consulting engagement. If there's any commonality at all across your engagements, however, your firm should be looking to create packaged IP. As this paper has described, what you create and how you'll monetize it depends on the specifics of the problem. **Figure 4** summarizes the choices.

Choosing the right option for your situation can be hard. If the need you've identified is too broad, you'll probably be facing competition from large existing ISVs (perhaps including Microsoft). If it's too small, you won't sell enough to make any money. What's required is correctly scoping the size of the problem, then creating the right kind of packaged IP to address it.

Whatever decision you make, the key point is this: The systems integration business is changing. While most SIs will still make a majority of their money from services for the foreseeable future, expect the mix to shift toward packaged IP. This transition appears to be inevitable, and so your best option is clear: get started now.

	Packaged IP	Typical Revenue Model	Market Characteristics	Example
Custom Projects	None	100% services	Unique solution	Creating SSAS-based data mining for a unique business
Custom Projects Built on an SI-Provided Foundation	Reusable frameworks	100% services	Repeatable, variable processes	Setting up an ETL pipeline with SSIS
Finished Solutions with Customization	Customizable application	30% services/ 70% software	Moderate number of customers with very similar needs	ERP reporting with Excel and Power BI for Office 365
Hosted Finished Solutions with Customization	Customizable application	Per-application, per-month payment	Moderate number of customers with very similar needs	Sentiment analysis with Windows Azure HDInsight
SaaS Solutions	Multi-tenant SaaS application	Per-user, per-month subscriptions	Many customers with nearly identical needs	Windows Azure-based access to and analysis of external data

Figure 4
Different packaged IP options imply different revenue models and market characteristics.

ABOUT THE AUTHOR

David Chappell is Principal of Chappell & Associates in San Francisco, California. Through his speaking, writing, and consulting, he helps people around the world understand, use, and make better decisions about new technologies.