Data Flows Made Easy

By Branden R. Williams
ISSA member,
North Texas, USA chapter

With data flowing all over the enterprise, it is no wonder that companies are having a hard time securing it. This article will explore a tool to simplify the creation and maintenance of data flow documentation.

With data flowing all over the enterprise, it is no wonder that companies are having a hard time securing it. Increasing amounts of information means increasing liability, and the increasing potential for mining the data to increase revenue. In the face of compliance, accurate data flows are imperative to an accurate compliance result. This article will explore a tool to simplify the creation and maintenance of data flow documentation.

Why is it that companies refuse to document their data flows? I'm not talking about the typical four-step diagram that says "First we swipe the card, then we authorize, then batch settle at the end of the day, and reconcile the next day." That one is utterly ubiquitous by now. And way too simple. I am referring to that deep dive to truly document, diagram, and understand data flows throughout the organization – that utopian diagram (that by definition does not exist anyway) that we all seem to yearn for, but never actually take steps to build.

Most security professionals agree that security is about protecting data. Yes, we implement many facets of security from policy to password controls, but ultimately we are trying to protect computing resources and the data that drives them. If that is the case, why do we resist (more like outright refuse) to validate our data flows and use them as tools in our everyday jobs?

As a consultant, I realize that many of the recommendations I make can appear "pie in the sky" to managers and analysts alike. A detailed data flow diagram is often something I suggest when giving recommendations for improvement. It should be that single encompassing document that is always updated and always reflects the marriage between design and reality, instantly showing how any change affects a company's security posture.

Even after urging repeat customers to consider this after another new finding surfaces that did not exist in the previous year, I still have not seen one implemented successfully. After thinking about the way that I was recommending this, I recognized a problem. Here is why companies are not doing detailed data flow diagrams (tell me if this sounds familiar):

This recommendation typically is communicated to C- or VP-levels in a company. They think it is a great idea, and pass it down to a Director to implement. The Director thinks it is a great idea and passes it to a Manager to implement. The Manager thinks it is a great idea, and places the dreaded "Hey John, I have a project for you" call to an unsuspecting Business Analyst.

"Hey John, here's a copy of Visio…Knock yourself out," says the Manager.

So John takes his copy of Visio, sits at his cube in Prairie Dog Land, and begins to realize the enormity of the project. This new project is so daunting that John does what most humans would do. He thinks something like, "I'll just work on answering email because I feel like I am accomplishing something by acting on and deleting emails in my Inbox. This project will just have to wait because people need things from me."

And there it sits, never to be completed.

But it is not the business analyst's fault! Rather, it is our fault as consultants for only providing an idea, and not a method to break the large task into smaller, more manageable tasks. In this article, I will present an adaptation of the Design Structure Matrix[1] that can be used first to validate that your designs match your implementation; then each interaction is numbered to make building a data flow diagram simple. Depending on how you organize the data created out of this adaptation, there are many ways to visualize it. The data organization is extremely simple, so any developer with any experience working with a database should be able to create a powerful visual front end.

This article will use examples relevant to individuals dealing with the Payment Card Industry Data Security Standard (PCI DSS), but pick any data flow that is critical to your business and this method applies.

How to map data flows

Step 1. Populate the Design Interface Matrix and Team Interaction Matrix.

Step 2. Merge your matrices to create your Alignment Matrix.

Step 3. Number your interactions in order.

Step 4. Build your visual flows by following the numbers!

Design structure matrix

For your initial pass you will likely want to use a spreadsheet to organize your findings. This will allow you to see an immediate visual representation, and it will be flexible enough to modify your structure while you discover more about your data flows.

One thing you will note very quickly is that if your payment flows are very complex, managing this in Excel can become cumbersome. If you have a large flow, you may need to break the flows down into different views. Think about how you view a complex network diagram. You start at a high and wide level, but "zoom in" to other areas and get more detail; all the way down to the system or component level. In this case, you may want to create matrices that single out unique flows such as Authorization, Settlement, Reconciliation, etc. These flows will appear to have duplicate components, but this is where a custom tool may become useful.

1 Manual E. Sosa, Steven D. Eppinger, and Craig M. Rowles. "Are Your Engineers Talking to One Another When They Should?" *Harvard Business Review*, Volume 85, Number 11 (November 2007): 133-142.

To start, you need to build what I call the Design Interface Matrix. To build this matrix, you will talk to your designers and/or architects. These are the folks who dreamed up and documented the plans for the payment flow. You will likely discover multiple people involved here from different teams as you start to follow the trail of credit card numbers. When you have identified all the components, you then need to document all the designed interactions that are supposed to happen according to the plan. Make each box where a data interaction occurs red (i.e., one component provides data to another). Colors will make a difference later, however the specific chosen color is not important. Just make sure they are consistent – Figure 1.

Figure 1 – Design Interface Matrix

		Providing Data				
		Card Swipe	POS Terminal	POS Controller	PayFlow	Bank
Receiving Data	Card Swipe					
	POS Terminal	1		7		
	POS Controller		2		6	
	PayFlow			3		5
	Bank				4	

You can number these if you like; however, you will renumber when you create your Alignment Matrix. I have included numbers on this graph for illustration purposes only. We are mapping data interactions for this simplified authorization data flow in this figure. Every application or component of the payment process is listed on both the horizontal and vertical axis, represented on the X-axis as Data Providers and the Y-axis as Data Receivers.

When we interviewed the designers, we noted that the normal authorization process starts with the Card Swipe device providing data to the POS Terminal (Box #1), then the Terminal providing data to the Controller (Box #2). The Controller in turn provides data to the PayFlow application (Box #3) and PayFlow provides information to the Bank (Box #4). At this point our data is coming BACK towards us, so you see that the Bank is providing data back to Payflow (Box #5) which then provides that data to the POS Controller (Box #6) and ultimately the POS Terminal (Box #7) so that the terminal knows how to continue with the transaction (authorized or denied).

Now we have the first part of our process, the Design Interface Matrix. This is how things are supposed to go. As in most of life, what looks good on paper rarely mirrors what happens in the real world. Designers can sometimes lose touch with reality and information provided to the implementation teams may not be sufficient to completely build out the solution.

Feet on the street sometimes need to be creative in their solutions, and that's where you will get variations.

Team interaction matrix

When it comes to any data security standard, variations in implementations can be costly. During an assessment, it is common to discover one particular team doing something that no one knows about. If you are trying to corral your compliance governed data into one specific set of systems to ensure proper controls exist, how devastating would it be to find out a compromise occurred on a system you did not know even had that data on it?

That is why the second activity we do is build out our Team Interaction Matrix. The data represented in this matrix is obtained from the teams actually implementing and managing the systems. These are the people you go to when something breaks, when you need some specific feature improvement, or when a change is required. We will call those folks the Implementation Teams. When we interview them, we find a significantly different picture – no surprise there! In virtually every assessment I have performed, this situation has occurred at least once: Team X describes an intricate process that no one but Team X seems to know about.

When filling this matrix out, you should have the same components on your X- and Y-axes that you had in the Design Interface Matrix. The difference is that you will use blue to fill in any interaction that exists when you talk to the Implementation Team – Figure 2.

Figure 2 – Team Interaction Matrix

Receiving Data \ Providing Data	Card Swipe	POS Terminal	POS Controller	PayFlow	Bank
Card Swipe					
POS Terminal					
POS Controller					
PayFlow					
Bank					

In our Team Interaction Matrix from our fictitious authorization process, we can see that what is happening in reality is actually quite different from what is supposed to be happening. If we went to the designers to build us a new process for something and they did not realize their designs had been altered, things might get out of hand.

Alignment Matrix

Once we have both matrices built, we can then merge them into what we call the Alignment Matrix – Figure 3.

Figure 3 – Alignment Matrix

Receiving Data \ Providing Data	Card Swipe	POS Terminal	POS Controller	PayFlow	Bank
Card Swipe					
POS Terminal					
POS Controller					
PayFlow					
Bank					

	N/A
	Unattended Interface
	Unintended Interface
	Matched Interface

Figure 4 – Key for Alignment Matrix

Per the key for the Alignment Matrix (see Figure 4), any area that your designers intended a data interface to exist and none is happening stays red, indicating an "Unattended Interface," or a data interaction that was designed to happen but is not being used in reality. For areas where the Implementation Team described a data exchange that is happening where the designers did not describe the same interaction, leave those boxes blue. This indicates an "Unintended Interface," or a data interaction that is happening today but was not designed to happen by the design team. Finally, where design meets reality, those boxes should be colored purple, indicating a "Matched Interface."

Let's review the Alignment Matrix to illustrate how this could happen. In our Alignment Matrix we see that the POS Terminal is providing data directly to PayFlow in addition to providing it to the POS Controller. Why would this occur? In the case of most companies, maybe there was a break-fix situation with the POS Controllers. Maybe they were unreliable during a critical retail day, or maybe there is some sort of rift between the Controller and Terminal development teams. In order to ensure that transactions continue to authorize, the POS Terminal team called the PayFlow team and asked if they could get an interface to provide transactions to PayFlow direct from the terminal. The PayFlow team says, "Sure" and now we have an alternate data flow.

While this may seem like a stretch, I have seen things like this happen in almost every company for which I have consulted. Immense pressure is put upon Implementation or Support teams to keep the systems running, and sometimes their creativity could get the best of your compliance situation.

For another example, let's say that the Controller and PayFlow teams are having some issues, and the Controller team asks the bank if they can pass transactions directly without going through PayFlow. The Bank says, "Sure," and now PayFlow is being bypassed! Let's say this method is so successful that the

teams stop relying on PayFlow altogether. Now data is being provided between Payflow and the Bank, but no one is using that data flow. This flow is supposed to be used, but is not due to some on-the-fly decision to bypass it: an Unattended Interface. Again, this is not as farfetched as it may seem and I bet it is happening in your company today!

So now that we have our Alignment Matrix, our job is to try to get the designers and the implementers on the same page. Once we do, we should not have any blue or red squares, just purple squares indicating that our designs match reality – Figure 5.

Figure 5 – Matched Alignment Matrix

Receiving Data \ Providing Data	Card Swipe	POS Terminal	POS Controller	PayFlow	Bank
Card Swipe					
POS Terminal					
POS Controller					
PayFlow					
Bank					

Now that we have a fully matched Alignment Matrix, it is time to start numbering the interactions as they happen. To illustrate a more complete data flow, I have mocked up a large matrix loosely based on an assessed company – Figure 6.

In this matrix, there are two different purple colors. This is not meant to confuse you, but to show you where you could have a parallel task occurring. For example, the eReconcile application is used to reconcile settled transactions with the bank in our matrix above. After it runs there are three parallel outputs that happen. Simultaneously, reports are sent to the Fraud Service, Marketing Data Collection,[2] and Reconcile Exception[3] applications. In each case, these reports essentially terminate the end of the data flow. Marking where flows terminate and originate is helpful for the person creating the diagrams and can also signify data stores that may need to be investigated.

Upkeep and value

Unfortunately this tool does have a flaw; it is susceptible to the "Garbage In, Garbage Out" problem. If you do not maintain it or keep the information up to date, you will never have an accurate picture from which to make decisions. Thus, validating your final Alignment Matrix (by building your Design Interface and Team Interaction Matrices, then merging) should be part of a quarterly self-assessment process.

As additional reinforcement, you should consider licensing some data discovery tools to make sure that sensitive data is not "leaking" outside of your expected systems. Common examples of this include those dreaded Excel spreadsheets or Access databases that can show up on laptops or desktops throughout the environment. Or, it could be an Unattended Interface that suddenly changes to include social security numbers.

The real magic here is how you implement the tool that stores this information. This is a conceptually simple process from a data perspective. A program simply needs to record the identifying components and the direction data flows between them. Anything added on top is gravy; such as the category that particular interaction is part of (Authorization, Settlement, etc.). Adding these tags could allow you to extract flows for specific applications or processes, or create animated drill-down maps.

Ultimately, you could get to a level of detail that might allow you to map flows within applications (continue drilling down into sub-component flows where appropriate; think web services). This might be necessary to properly document flows in support of audit activities.

The audit process

What was that you said? Did you say I could use this to help me with that dreaded audit process? I sure did! I have been on both sides of the fence of auditing activities. One of the most time consuming chores is educating outsiders (even internal audit can feel like an outsider sometimes) on how your back-office technology works. For example, when looking for data subject to HIPAA, imagine how much time you could save by pulling out detailed data flow diagrams and the matrices shown here to quickly illustrate how your process works.

Not only does this illustrate your data flows, but gives you a high confidence that what you present is exactly what outsiders will find if they start poking around!

Providing this level of concise detail also yields immediate benefits by building management confidence in knowing your compliance posture (know the outcome of an assessment before it begins), reducing the overall effort (i.e., cost) an audit might require, and minimizing the resources that are consumed by auditors in meetings.

In a meeting with a colleague recently, it occurred to me that these diagrams could totally replace those dreadful Visio flow diagrams that we are programmed to create and decipher. Through this method, a member of my team was able to take a detailed Visio diagram that nearly required a magnifying glass to read and reduce it to a simple 8x8 matrix as seen in Figure 6 above.

Better informed decisions

What other benefits can be extracted from this process? How about having a group of people in your company that actually know as much, or even more than external auditors about

2 An application used by the marketing folks to collect data on customers.

3 A report showing where transactions did not correctly reconcile with the bank.

Figure 6 – Full Matrix for Complete Payment Flow

	Card Swipe	POS Terminal	POS Controller (Auth)	POS Controller (Settle)	PayFlow (Auth)	eSettlement (Settle)	Bank (Auth)	Bank (Settle)	Bank (Reconcile)	eReconcile	Enterprise GL	Fraud Service	Mktng Data Collect	eChargeback	Reconcile Exception	Bank Chargeback	Data Warehouse
Card Swipe																	
POS Terminal	1		9														
POS Controller (Auth)		2			8												
POS Controller (Settle)																	
PayFlow (Auth)			3				7					5					
eSettlement (Settle)				10													
Bank (Auth)					6												
Bank (Settle)						11											
Bank (Reconcile)																	
eReconcile									12								
Enterprise GL																	
Fraud Service					4					13							18
Mktng Data Collect										13							18
eChargeback																15	17
Reconcile Exception										13							
Bank Chargeback																	
Data Warehouse													14	16			

your data flows from end to end! Rarely do we find any one person who can describe, in detail, data flows from end to end. Having this data available allows you to make better, faster decisions with more confidence that you will not run into snags down the road.

Regardless of how you approach it, accurate data flows are imperative to companies who are charged with protecting sensitive data.

About the Author

Branden R. Williams, CISSP, CISM, has fourteen years of IT experience, the majority concentrating in information security. He is currently the Director of the PCI Consulting Practice at VeriSign and regularly consults with top global retailers, financial institutions, and multinationals on enabling secure business growth. He can be reached at bwilliams@verisign.com.