

DOCUMENT ENGINEERING

Robert Glushko, Tim McGrath, & Brian Hayes

Chapter 0. Introduction to Document Engineering

A Simple Business Transaction?

Imagine that you go into your local "bricks and mortar" bookstore and notice a new book with an intriguing title, "Document Engineering" by Glushko, McGrath, & Hayes. You hand the clerk your credit card, and a few moments later you leave the store with your copy of "Document Engineering." If you were to describe what had just taken place, you might say that you "purchased" a book, using just the single word "purchase" because the experience certainly seemed like a single economic event or transaction taking place between you and the bookstore.

Now imagine that you are browsing the web site at an online bookstore, GMHBooks.com. You navigate a few menus to select a new book with an intriguing title, "Document Engineering" by Glushko, McGrath, & Hayes. You type in your credit card information and your address, and a few days later the book arrives by delivery service.

How would you describe what took place at GMHBooks.com? At first glance the online experience seems equivalent to the "bricks and mortar" experience, so you might describe it as "purchasing a book online." But if we analyze the online experience more closely, we can identify at least three separate "transactions" or information exchanges that occurred:

1. Your interaction with the GMHBooks.com catalog to select the book you want to order;
2. An information exchange between GMHBooks.com and a credit card authorization service (like VISA or MasterCard) to verify your creditworthiness and to charge your account,¹
3. An information exchange between GMHBooks.com and the delivery service with the instructions for picking up and delivering your book.

So what looked at first like the single event "buying a book" turns out to be at least three separate events that have been combined in a particular sequence or business process. And as we fill in some more details to make the example more complete, we'll see even more transactions.

For example, unlike the bookstore in the physical world, GMHBooks.com doesn't have its own inventory of most of the books it offers in its catalog. Instead, it maintains a "virtual inventory" in its catalog, which consists of the books it can quickly obtain from book distributors when a customer selects them from the virtual catalog. So other transactions that might take place are:

¹ This validation might have taken place in the "bricks and mortar" bookstore, too, but it is essential in the online store because more stringent regulations apply when you enter the credit card online and the merchant doesn't see the actual card

4. An information exchange between GMHBooks.com and the distributor to confirm that the book you selected is available so that GMHBooks.com can sell it to you and promise a delivery date.
5. The order sent by GMHBooks.com to the distributor to obtain the book on your behalf.

This simple example contrasting a "bricks and mortar" bookstore and an online bookstore illustrates the disruptive force of the Internet on traditional business models. Despite their superficial similarities, the two stores are radically different. Unlike the physical bookstore, which exists as a single entity, the online bookstore GMHBooks.com is a "virtual enterprise" that emerges from the coordination of the activities of numerous independent businesses that collaborate to achieve their interlocking goals. This coordination takes place via the exchange of information, most often in the form of electronic messages or documents. Efficient coordination with book distributors and shippers enables GMHBooks.com to do without its own books and delivery trucks – it can replace inventory and equipment with information.

Introducing "Document Engineering"

Internet communities, public and private marketplaces, auctions, information brokerages and syndication networks are all examples of business models that combine or interconnect products or services from multiple businesses. All are possible because of the Internet's efficiency for exchanging information and documents. All of them rely on the premise that what matters most when two businesses or applications exchange information is that they agree on the meaning and purpose of the information. That is, the businesses must agree on the documents that serve as interfaces to their respective processes. They need not agree on or even know anything about how those processes are carried out or the technology that implements them but they must agree on the semantics of the documents they exchange.

Using document exchange as a technology-independent abstraction is essential and natural in the heterogeneous information technology environment of the Internet. Using documents as interfaces is the basis of the excitement about "web services" as a way to create new applications or expose existing ones in a manner that enables them to be easily combined or integrated. The "loosely-coupled" architecture of document exchange reduces the cost of designing and implementing new applications, which is essential as businesses experiment with business models that take advantage of the Internet.

But where will all of these electronic documents needed for web services, information supply chains, and virtual enterprises come from? "Document Engineering" is evolving as a new discipline for specifying, designing, and implementing the electronic documents that use Internet information exchange mechanisms to request or return the results of business processes. The essence of Document Engineering is the analysis and design methods that yield formal models to describe the information these processes require and the sequence or choreography by which related processes are coordinated.

Analysis Perspectives and Patterns in Document Engineering

There are many different perspectives from which we might analyze and describe the document exchanges in even a simple scenario like buying a book at the online bookstore GMHBooks.com.

At the most general or abstract level, a pattern of document exchanges can be viewed as an analogue to a class of business models. Conceiving of business models in terms of document exchanges makes it easier to align business strategy and technology architecture and easier to understand the constraints imposed by legacy systems. Different business models can imply distinct patterns in inter- and intra-company coordination as implemented in document exchanges.

For example, when a company sees a market opportunity for a new kind of product, one business model might have it designing, manufacturing and distributing the product itself as a completely vertically integrated enterprise. This business model implies very tight coupling of the information systems within the firm (often with MRP or ERP systems) with relatively few information exchanges with outside firms. Alternatively, the business model might involve assembling the product from subcomponents produced by other suppliers, which requires extensive information sharing and very efficient coordination with the companies in the supply chain, especially if the product is sold to customers in a customized or build-to-order fashion. Alternatively, after designing the product, the company might have it manufactured and distributed by other firms, which may require significantly fewer and less granular information exchanges than in the component assembly model.

This high-level perspective of the business model provides a qualitative view on document exchanges that can be very useful in experimenting with new business models or applying them to new domains. The abstractions and patterns in business models that are most useful in document engineering are described in Chapter 1, "Patterns in Business."

In the online bookstore example, the business model used by GMHBooks.com can be described as "drop shipment," where an inventory-less retailer offers products from an aggregated catalog and routes customer orders to publishers or other firms who fulfill the orders from their own inventories. While this coordination is usually invisible or appears seamless from the customer's perspective, it requires a complex choreography of document exchanges. Many "dot coms" failed because their flashy web sites were able to take orders from customers but did not implement the "back end" information exchanges with warehouses and shippers required to make delivery promises to customers that could be met. Dissatisfied customers didn't return.

A more detailed view of the document exchanges and business processes than that provided at the business model level is justified in most situations. We need to dig deeper to identify the individual transactions and the roles they play in the overall business process, as we did in the GMHBooks.com scenario. It is essential to understand and carefully describe the patterns of document exchange in a business process because the businesses at each end of the exchange must agree on them to do business, and the same documents can be reused in different business contexts that follow different patterns of document exchange.

The need for agreement is obvious: what if GMHBooks.com sends a Purchase Order to a book distributor and expects a Purchase Order Acknowledgment in return, but the book distributor's normal business process when it can fulfill an order is to send an Invoice and a Shipping Notice? GMHBooks.com's information systems might be incapable of handling the electronic response from the book distributor and the order would be lost. Instead, GMHBooks.com might accompany the order with a process specification document that proposes a sequence of document exchanges for handling purchase orders. If the two firms are using "business process management" software, they might use it to align their business interaction by ensuring that the appropriate documents are exchanged and processed in the specified sequence.

Techniques for analyzing and designing business transactions and how they combine in business processes are covered in Chapter 6, "Business Process Analysis." Methods for recording this analysis in business process models are described in Chapter 7, "Business Process Models."

Understanding the processes tells us HOW we doing business exchanges, but not WHAT we are exchanging. In addition to analyzing and describing the pattern of document exchanges in transactions and processes, it is essential to look inside the documents to understand their content. It is usually the case that much of the information "flows" from document to document– or stays the same, depending on how you look at it. In the online bookstore example, information about the selected book flows from the Catalog to the Order and on to the Packing Slip or Manifest that arrives with the book; likewise, the buyer's Address may be reused in numerous documents.

The reuse of information in a set of related documents is facilitated by structural patterns. The most important of these is that the documents are all composed or assembled in "building block" fashion from smaller logical components like Party or Item or Address. Reuse of these components is facilitated when their semantics conform to standards, which have been developed for decades for specific vertical industries by trade associations, industry consortia, or formal standards bodies. "Horizontal" standards for logical components needed in every industry are a more recent development but are no less important. For example, descriptions of businesses and individuals, basic item details, measurements, date and time, location, country codes, currencies, business classification codes, and similar "atomic" or "primitive" information units are needed in every industry in a wide variety of documents.

A secondary type of document pattern may be based on technical requirements such as ease of processing or human readability. Particular assembly structures, such as a "header – details – summary" may be shared by several documents (most often by paired documents that play request-reply roles, like Order and Order Acknowledgment).

Techniques for analyzing and designing documents and other information patterns are described in Chapter 8, "Information Analysis." Methods for recording this information analysis in models are covered in Chapter 9, "Information Models."

Implementation in Document Engineering

Analyzing the patterns in documents and the patterns of document exchanges that support business processes are critical steps in designing or understanding how a business

model works. But to make this knowledge fully interoperable between applications it must be encoded in a computer-processable form, and that means XML. XML – the Extensible Markup Language – is being touted as the lingua franca of the Internet. HTML's fixed tag set is not expressive enough, and the EDI formats used by legacy applications are not flexible enough to handle inter-application communication between a wide range of ever-changing companies and services.

XML and these other technologies for encoding and exchanging documents are described in Chapter 2, "Technology Foundations."

But nothing about XML per se ensures that XML documents and process specifications needed by Web Services and other applications are meaningful or even mutually intelligible. Since XML has no fixed semantics, it can be used to describe anything. But this means that XML doesn't come "out of the box" with a standard way to describe anything. Using the same tags isn't sufficient, either. The same content will invariably be described using different names (an Address element in one document might mean the same as BuyerAddress in another document), and different content might be given the same name (Address might mean BuyerAddress in one document and SellerAddress in another).

This requirement for well-defined semantics in XML models trumps all other considerations; it does no good for a business to expose an application interface and agree to accept an XML document that embodies it if no other business can be sure it understands it or the XML document that it gets back. This fundamental "interoperability challenge" and techniques for resolving it are described in Chapter 3. Using XML -- and especially XML standards -- to encode conceptual models of documents and business processes to give them a rigor, reusability and interoperability lacking in other representations is the subject of Chapter 10, "Implementing Models."

Encoding document and process models in XML gives a common view to information aggregation, management and distribution. XML from documents, databases, and enterprise applications can be combined and treated as a single source from which content can be transformed for multiple users, devices, or applications. XML-centric programming paradigms are emerging in which XML schemas, programming language objects, database schemas, and UML models can be treated as equivalent. The techniques and architectures by which XML is a unifying technology for implementing applications that use Internet protocols and web service standards, and especially for those that span enterprise boundaries, is the topic of Chapter 11, "Architecture and Deployment."

Nonetheless, the emphasis on XML in document engineering is less fundamental than the philosophy or business model that it serves – that different organizations, applications, information sources, and devices can function as cooperating entities regardless of where they fit within or across enterprise boundaries. XML cannot achieve the compelling vision of the "virtual enterprise" or the "plug and play economy" without consideration of management and strategic issues. These are covered in Chapter 12, "Management and Strategy."

Chapter Summary

Using document exchange as a technology-independent abstraction is essential and natural in the heterogeneous information technology environment of the Internet. The "loosely-coupled" architecture of document exchange reduces the cost of designing and implementing new applications, which is essential as businesses experiment with business models that take advantage of the Internet.

"Document Engineering" is evolving as a new discipline for specifying, designing, and implementing the electronic documents that use Internet information exchange mechanisms to request or return the results of business processes. The essence of Document Engineering is the analysis and design methods that yield formal models to describe the information these processes require and the sequence or choreography by which related processes are coordinated.

Good Document Engineering practice emphasizes the reuse of existing models wherever possible and requires that new models be described in ways that encourage their reuse by others. Reusable patterns in Document Engineering can be found at the conceptual level in terms of libraries of models that describe common business processes and the organization of activities between businesses. Conceiving of business models in terms of document exchanges makes it easier to align business strategy and technology architecture and easier to understand the constraints imposed by legacy systems. In addition to analyzing and describing the pattern of document exchanges in transactions and processes, it is essential to look inside the documents to understand their content. It is usually the case that much of the information "flows" from document to document— or stays the same, depending on how you look at it. Modeling this reuse of information within a set of related documents is best accomplished with reusable patterns in the form of XML libraries and standards.

XML-centric programming paradigms are emerging in which XML schemas, programming language objects, database schemas, and UML models can be treated as equivalent. XML has become a unifying technology for implementing applications that use Internet protocols and web service standards, especially for those that span enterprise boundaries.