The U.S. Treasury Tests A New Payment Mechanism

Ulric J. Gelinas, Jr.
Department of Accountancy
Bentley College
175 Forest Street
Waltham, MA, USA 02452-4705
ugelinas@bentley.edu

Janis L. Gogan
Department of Computer Information Systems
Bentley College
175 Forest Street
Waltham, MA, USA 02452-4705
jgogan@bentley.edu

Chuck Wade
Interisle Consulting Group
4 Tiffany Trail
Hopkinton, MA 01748-1630
Chuck@Interisle-Group.com

ABSTRACT

This case presents a set of technical issues confronting the United States Treasury eCheck Pilot Project team in January 2000. The team, which included representatives from the U.S. Treasury, the Federal Reserve Bank of Boston, Fleet Boston, Bank of America, and several hardware and software vendors, was testing a new Internet-based payment mechanism (eCheck). The system had already been tested for a year and a half with the participation of the two commercial banks (Fleet Boston, Bank of America), but this portion of the pilot was now coming to an end. During the first phase of the project, several key design choices had been made, including the use of smart cards to hold digital certificates, and specification of the information flows among the participants (payer, payee, payer bank, payee bank). Now, the system would need to be modified so that the U.S. Treasury could continue to make eCheck payments to a few defense contractors, with the help of the Federal Reserve Bank of Boston. Two new designs are presented for evaluation.

Keywords: eCheck, Internet, payment mechanisms, systems design, emerging technologies

1. INTRODUCTION

In January 2000, the United States Treasury eCheck Pilot Project team was planning the next phase of this test of a new electronic payment mechanism, which involved participation from the Treasury's Financial Management Service, the U.S. Department of Defense Finance and Accounting Services Division, the Federal Reserve Bank of Boston, and a few Defense suppliers. An earlier phase of the project had also involved two commercial banks, Fleet Boston and Bank of America, but this next phase would not include commercial banks. Thus it was necessary to redesign the payment flows. One solution had been suggested by Frank Jaffe, the outgoing manager of the eCheck Pilot Test. Another solution was suggested by Mike Versace, from the Federal Reserve Bank.

Participants on the eCheck team–especially the representative from the U.S. Treasury Financial Management Service and Mike Versace from the Fed–needed to decide which of these two approaches to take.

2. eCHECK PROJECT BACKGROUND

eCheck was one of several projects initiated by the Financial Services Technology Consortium (FSTC), which consisted

of financial institutions, hardware and software firms, governmental agencies and others. The eCheck project, begun in spring 1994, was aimed at developing a new electronic payment mechanism for use in Internet commerce and other contexts. A Proof-of-Concept demonstration was held in 1995, and in 1996 a decision was made to conduct a pilot test at the United States Treasury (the decision was not announced until fall 1997, after all parties signed project contracts). Much work was then done to flesh out the detailed specifications for ensuring secure transactions before the first eCheck was cut on June 30, 1998.

This case describes the evolution of the eCheck design and technical specifications through winter 2000. A companion case (Gogan, Gelinas and Rao, 2003) addresses strategic and project management issues.

3. PROJECT PARTICIPANTS

The pilot project was officially announced on October 7, 1997. Participants (listed in Exhibit 1) had expected that the pilot test would involve 50 vendors, run for one year, and process up to 1,000 checks and $1 million per day. But before payments could be made, several important design issues had to be resolved. The next three sections of the case discuss each of these design challenges.

4. SMART CARD DESIGN: A TOKEN CHOICE?

Early on, security experts on the eCheck design team (such as Milt Anderson, a cryptography expert from Bellcore, Ken Goldman, a security researcher at IBM, Doug Kozlay, a founder of Information Resources Engineering (IRE), and Chuck Wade, a specialist in PKI services at BBN) urged the use of a separate "token" for storing cryptographic private keys (a "security token" is a simple hardware device, such as a smart card, key fob or small keypad, that is used in conjunction with another hardware device). A user would need to insert a specially designed card or device into a reader on their computer, before an eCheck could be digitally signed and sent on to the payee. Milt Anderson explained:

> "In two-factor authentication, the user must *have* something–a token–and *know* something–a password. If I leave my laptop PC at the airport, I'll have plenty to worry about, but my eCheck account will be safe. If my eCheck smart card falls into the wrong hands, that's okay as long as the thief doesn't know my password. If I carelessly reveal my password, then the thief must obtain my smart card, which imposes one more security hurdle for the bad guys to surmount."

Some participants questioned the choice to store security keys on smart cards. Frank Jaffe, who represented Bank Boston and also served as overall eCheck project manager, argued in favor of a simpler approach:

> "Not all computers have PCMCIA slots, and I'm no longer convinced a token is necessary, from a business perspective. Another approach: store the key for the user's digital signature in an encrypted file on their hard drive… This is not quite as secure as a smart card, but … it's good enough. Most firms use firewalls to prevent unauthorized penetration. There is always a trade-off between perfect security and usability. Since it's fairly unlikely that bad guys can obtain digital signature keys on a large scale, it's more practical to establish just-in-case corrective procedures for the unlikely event keys are compromised."

Security experts on the team argued that the use of a hardware token is inherently more secure than a software solution. After much discussion the team chose to err on the side of caution and utilize a PCMCIA card. At the time (1995) the team was told that virtually all PCs would be equipped with PCMCIA slots by 1998. But as of 2000 few desktop models had a PCMCIA slot (an external PCMCIA device cost about $60), although most laptops did. Meanwhile, by 1997 an alternative token was gaining ground: the so-called "smart card," which contains an embed-ded processor. Initially it was felt that smart cards were not powerful enough, but IRE and others on the team coded the necessary function-ality into a new generation smart card which was less expensive than a PCMCIA card (and, external smart-card readers cost only about $20). Team members agreed that a reasonable comp-romise between cost and security was achieved.

5. HOW TO STUCTURE ECHECK DOCUMENTS?

A markup language defines how information will be presented on an output device (such as a screen), and how portions of a document can serve as input to appli-cation programs. SGML (Standardized Generalized Markup Language) is a set of specifications for creating a markup language. HyperText Markup Language (HTML), used to display web documents, is defined by SGML. In 1994 the eCheck designers evaluated the use of a markup language to structure eCheck documents. Some problems with extant markup languages were identified, especially when a necessary requirement was to digitally sign eChecks using public key cryptography. So, the team developed a new SGML-compatible mark-up language, for financial applications only, and eCheck in particular: Financial Services Markup Language (FSML). Milt Anderson noted: "The simplicity of FSML makes it compatible with the memory, process-sing, and interface speed limitations of smart cards." FSTC published the specifications for FSML in fall 1998. That year, the World Wide Web Consortium approved Extensible Markup Language (XML) Version 1.0. XML meta-tags provide "information about infor-mation" (i.e., tags indicate what type of information is in the document), an approach that was similarly used in FSML. The general-ized use of meta-tags in XML made it possible to design a broad range of XML-compatible applications. Had XML already been an established protocol just two

years earlier, the eCheck team would surely have given it serious consideration (although FSML was a better fit for the eCheck application, having been custom designed for that purpose). By 2000, XML was a widely accepted industry standard. The eCheck team planned to re-write FSML as a subset of XML, but no deadline had been set.

It is worth noting also that XML presented a significant technical hurdle for eCheck applications. Most markup languages inherit from SGML a limitation that is not found in other computer languages. Specifically, most markup languages define internal names (*e.g.,* variables) as "global," meaning that the same name must be unique throughout a document. When two documents are combined into a new document (not uncommon, and vital for the eCheck application), name collisions must be resolved by defining replacement names that will be unique in the combined document. If documents have been digitally signed, their names cannot be redefined, since any change to a document invalidates the signature (a useful feature if you don't want someone modifying the amount field in an eCheck after it has been signed). This problem was resolved elegantly in FSML by defining name scoping rules (similar to how most computer languages work). An FSML name need be unique only within a subdocument that contains the name. Since XML lacks this feature, digitally signing XML documents was a significant challenge that required a few years of additional work. The FSTC provided requirements that provided a foundation for an IETF/W3C joint working group, known as XML DSig.

6. FOUR CORNER ECHECK MODEL OR ECHECK LOCKBOX MODEL?

A key design choice was determining how communication would take place between payer and payee. The eCheck team pioneered a method for using secure email transmissions for eCheck itself and for related remittance information. Two processing models were evaluated. A "four-corner" model (Exhibit 2) was proposed in fall 1995; Treasury, the Fed, and Nations Bank approved this approach. Subsequently Frank Jaffe at Bank Boston advocated another approach. Instead of fully paralleling paper check processing (see Exhibit 3), Jaffe preferred to process eChecks in a manner similar to a lockbox (Exhibit 4). He argued vigorously in favor of the lockbox (Exhibit 5) approach, stating:

> "The [traditional] four-corner model is far and away the hardest one to implement, because all four parties–payer, payee, payee's bank and payer's bank–have to get equipped before you can flow your first transaction through."

The project manager at NationsBank, Steve Schutze, had a different point of view:

> "I see why you're in favor of using a lock-box, Frank, because it's easier. But with a lock-box you must convert the eCheck to ACH format and you give up a lot. You no longer have any of the attributes of a check when it gets posted to a recipient's ac-count. If we're going to call it a check, we should process it as a check."

Treasury and the Fed agreed with Steve Schutze that the Pilot should test the peer-to-peer capabilities of eCheck (with an eCheck being directly sent to the recipient, just like a paper check). This choice meant that the NationsBank eCheck server would need to be able to encapsulate eChecks into a check image format (using X9.46, a standard that governs the electronic exchange of digital images of checks).

Mike Versace preferred that eCheck utilize the Federal Reserve Bank's existing systems and network infra-structure as much as possible (much of which was only recently deployed). The Fed's server would also apply digital signatures and certifi-cates. In order to submit payments for clearing through the existing Electronic Check Presentment (ECP) system, the Fed's server would "wrap" each eCheck "image" in a secure enclosure (using another standard, X9.37, which specifies how to enclose payment data for clearing).

The U.S. Treasury wanted Treasury check law to apply to the eCheck and this would not be the case with the lockbox model. So, they too wanted eCheck to behave like a paper check and be processed like a paper check. They wanted the trial to test a "peer-to-peer" model of payment, which is comparable to traditional paper check flows (i.e., an eCheck is delivered directly to the payee, who logs its receipt and then deposits it at their bank).[1]

Once participants agreed to use the traditional "around the outside" four-corner processing model, it was possible to complete the necessary server software. On June 30, 1998, Treasury cut the first eCheck, for $32,135.97 to pay GTE for work on a Defense Department contract. The check was cleared through the Federal Reserve Bank of Boston and deposited into GTE's account at BankBoston.

7. MEDIATED Z-FLOW OR INTERNET PAYMENT SERVICE MEDIATOR?

BankBoston and NationsBank committed to participate in a 12-month pilot (Fall 1997 – Fall 1998). In fall 1998 NationsBank merged with Bank of America, but Bank of America agreed to an extension of the pilot (as did BankBoston). In fall 1999 BankBoston was acquired by Fleet Bank. Bank of America then announced that they would withdraw from the pilot in spring 2000. Since the pilot design required participation of two commercial banks, Fleet

[1] If eCheck were used for other than Treasury checks, Uniform Commercial Code sections 3 and 4 (check law) would apply only if eChecks were processed in the same manner as paper checks, and not converted to other payment systems (such as ACH, which is governed by different laws and industry rules).

Bank had to withdraw from the pilot.[2] By then, Fleet had decided to spin off its eCheck interest into a new venture, Clareon. Frank Jaffe decided to join Clareon, so he would soon leave the eCheck team).

For the next phase of the Treasury Pilot, the team needed to determine how to replace Bank of America and Fleet as the vendors' (payees') banks. After some ideas were discussed, Mike Versace at the Fed offered to have vendors deposit their eChecks at the Fed, which would then clear them through the Automated Clearing House (see http://nacha.org/About/what_is_ach_.htm); He explained (see Exhibit 6):

> "Defense Department vendors will still receive eChecks from Treasury. But with this new flow, vendors will send their eChecks to the Fed. We will then convert the eChecks to ACH credits to transfer the funds from Treasury to the vendor."

Versace added that the systems at the Fed would "figure out where to send the money through the use of an intelligent interface to the ACH network." He was quite enthusiastic about this approach:

> "This clearinghouse model represents a different way of thinking for the Fed. It puts us in a new intermediary position–between vendors and their banks. And, it gives us the opportunity to learn more and extend our support for the Treasury."

Before making a final decision, the team asked Frank Jaffe for his opinion. Jaffe liked the idea. He replied:

> "So, the Fed will become an originator of ACH credits rather than clearing the payment as a check. This approach is a better model for the eCheck. The [traditional] four-corner model should be dumped. It was done as a technology proof-of-concept, but as a business model implementation it's the hardest way to go about doing this and getting adoption, because too many different parties need to be enabled."

"Still," he added, "one could go even further." Jaffe laid out yet another way to design the information flows (see Exhibit 7), using a central Internet Payment Service:

> "The right way to get adoption is to have a processing service–be it a bank or not–to do just-in-time application delivery. This would be a service, not a technology or a software solu-tion. There would be no customer-side install, and no special software at the bank. The service would provide a raw ACH file to the bank.**"**

Jaffe's design eliminated the need to install special server software at participating banks, so he believed that bankers would be enthusiastic about trying this new payment service. Jaffe would also revisit the decision to use two-factor token-based authentication in this scenario. "I'd issue passwords to users and design plenty of security into the central service," he explained. "I would not require users to insert a card into a slot every time they want to make a payment."

With the rapidly approaching withdrawal of Fleet and Bank of America), in January 2000 it was time to decide on a new processing flow. As they considered the merits of Frank Jaffe's Internet payment service versus the Fed's Z-Flow proposal, team members recalled that Milt Anderson (the security expert from Bellcore) believed that an electronic check could serve a greater variety of transaction types and trading partners, more conven-iently and at a lower cost, than any other payment mechanism. At the 1995 proof-of-concept demonstration he had said (Gelinas and Gogan, 1997):

> "eCheck is a message that tells existing demand deposit accounting systems to do credits and debits against existing systems. Encrypted digital signatures will authenticate banks and customer accounts. The code for producing and validating digital signatures will reside on customers' eCheckbooks (tokens), and banks' eCheck servers. Because an online intermediary won't be required to complete a transaction, processing costs will be lower than secure credit-cards or network money systems."

By January 2000 $3 million in payments had been issued in the eCheck pilot; the largest single payment had been for $230,000. The team considered: which of the two designs best matches the intent of the original designers? More importantly, which design represents the best solution as eCheck moves out of the Treasury Pilot and on to commercialization?

8. ACKNOWLEDGEMENTS

The authors thank members of FSTC and the U.S. Treasury eCheck Market Trial team, who generously gave time for interviews, provided documentation, and reviewed case drafts to verify facts. Our study received financial and in-kind support from Bentley College, and was conducted under the auspices of the Bentley College InVision Project (Jane Fedorowicz, Principal Investigator), whose members include: Ulric J. Gelinas, Janis L. Gogan, Phillip G. Knutel, M. Lynne Markus, Amy W. Ray, Catherine A. Usoff and Christine B. Williams.

9. REFERENCES

Gelinas, U.J. and J.L. Gogan. The FSTC electronic check project. Case no. 96-10, *American Institute of Certified Public Accountants Academic and Career Development*

[2] Another significant change was pending: FSTC signed a memo of understanding to transfer the eCheck technology to CommerceNet (another consortium) in January 2000. The arrangement was never fully consummated, and FSTC later re-established its stewardship of eCheck.

Division Case Development Program, 1997. www.aicpa.org/members/div/career/edu/caselist.htm

Gogan, J. L., U.J. Gelinas, and A. Rao. Is this pilot test over? *Annals of Cases on Information Technology*, 2003.

AUTHOR BIOGRAPHIES

Ulric J. (Joe) Gelinas, Jr., received his M.B.A. and Ph.D. degrees from the University of Massachusetts. He is co-author of *Accounting Information Systems, 5th ed., Business Processes and Information Technology*, and founding editor of the *Journal of Accounting and Computers*. He participated in the development of *Control Objectives for Information and Related Technology (COBIT)* by participating in the COBIT Expert Review and by authoring portions of the *Implementation Tool Set*. He is a recipient of the Innovation in Auditing and Assurance Education Award from the American Accounting Association. Dr. Gelinas has published articles and case studies in *Issues in Accounting Education, Information Systems Audit & Control Journal, Technical Communications Quarterly, IEEE Transactions on Professional Communication, Annals of Cases on Information Technology* and other outlets.

Janis L. Gogan is a member of the Bentley College CIS faculty, holds *EdM, MBA and DBA degrees from Harvard University.* Her teaching interests include management of information technology, electronic commerce, and IT project management. She has conducted research on the Internet as a disruptive technology, the management of emerging technologies, IT project management, and inter-organizational knowledge sharing. Dr. Gogan has published in *Communications of the Association for Information Systems, International Journal of Electronic Commerce, Journal of Information Technology Cases and Applications, Journal of Management Information Systems*, and other journals. For several years Dr. Gogan contributed a column, titled "Benchmarks," for *Information Week.* She has written numerous Harvard Business School cases, including several best-sellers which have been taught in MBA programs in the U.S, Europe, and Asia. She is a frequent speaker on the strategic implications of emerging IT issues.

Chuck Wade is a founding principal at Interisle Consulting Group. He holds both Sc.B. and Sc.M. degrees from Brown University in Electrical Engineering. He has substantial experience with complex distributed systems that deliver high availability, high performance and strong security. He consults in the areas of resilient system architecture, eCommerce security and Internet payments. He has served recently as a Senior Researcher at CommerceNet, and as a Principal Consultant in the Information Security Group of BBN Technologies. At BBN, he led Electronic Commerce initiatives and client engagements, with most of his consulting work within the Financial Industry. As one of the original participants in the FSTC eCheck Project, Mr. Wade has been involved with over-the-Internet electronic payments since the mid 1990's. He also contributed directly to the architecture, design, deployment and testing of various large, mission-critical networks, including the trading floor network for one of the world's most important Stock Exchanges. Mr. Wade spent all of the '90s with BBN (now a part of Verizon) as a Consultant and Systems Architect. During most of the '80s, he worked at Motorola directing the Advanced Technology Group for their Codex division. He has also worked in the minicomputer industry and university research.

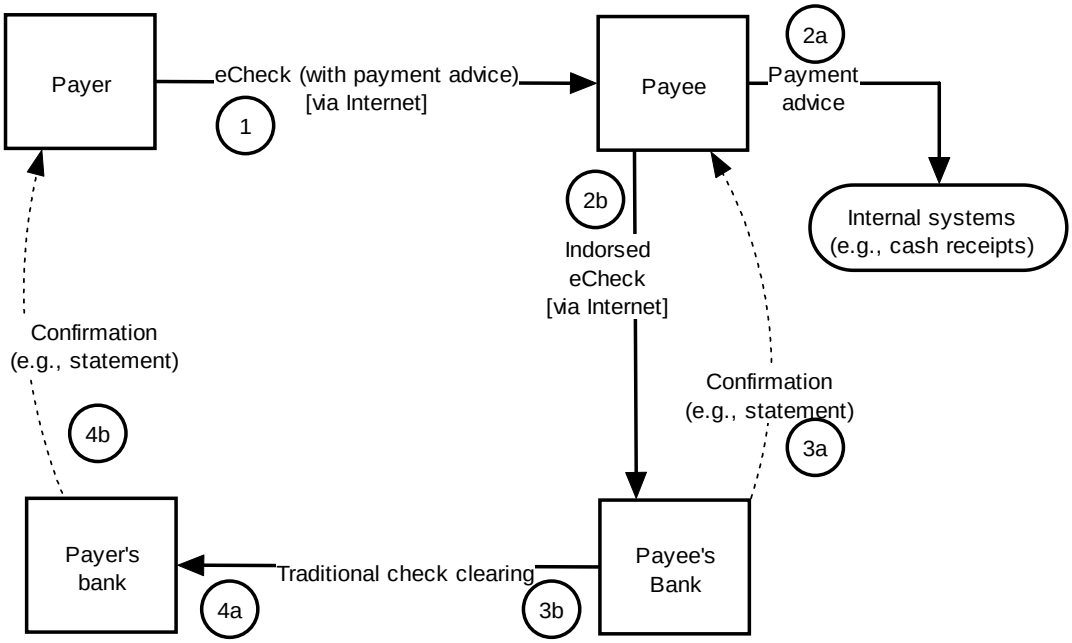
Exhibit 1
eCheck United States Treasury Pilot Participants

Participant	Role
BBN (GTE Internetworking)	Provide certificate authority software and hardware, plus high-assurance cryptographic hardware used by Treasury to sign eChecks. Chuck Wade initially managed BBN's part of the project.
BankBoston (later Fleet Boston) and NationsBank (later Bank of America)	Depository banks for the Department of Defense vendors. Heretofore, Frank Jaffe managed the overall eCheck project, while also managing Bank Boston's part of the project. NationsBank's part of the project was managed by Steve Schutze.
Federal Reserve Banks of Boston and Richmond	Clearing bank for U. S. Treasury. Mike Versace, who headed the Emerging Payment System Group in the Fed's Retail Payments Office, managed the Fed's part of the project, including end-to-end system testing (with participation by the Dallas Federal Reserve).
IBM, with Agorics	Develop eCheck servers for BankBoston and NationsBank, to accept eChecks via email and then process them through existing systems.
Information Resources Engineering (IRE):	Develop smart-card technology and integrate with RDM software.
IntraNet	Develop software for converting eCheck data to X9.46 standard[3]
RDM Corporation	Develop eCheck servers for Treasury and software for payees (Defense Department suppliers) to receive eChecks and submit eCheck data to their own accounting systems.
Sun Microsystems	Develop the eCheck servers for the Federal Reserve Bank
U. S. Dep't. of Defense Finance and Accounting Services Division	Actual provider of payments to Defense suppliers (by longstanding government policy, Defense made its own payments, in contrast to most federal agencies, whose payments are made by Treasury, although in this pilot, Treasury did act as the payments agent for Defense).

[3] In fall 1999 X9.46, was amended by ECCHO to include the eCheck image. X9.46 is a standard for electronic exchange of digital images of checks. This format was used for submission of eChecks to the Fed in Phase I of the pilot. At the Fed this image was enclosed in an X9.37 format for clearing. The X9B group of the Accredited Standards Committee, Inc. X9—Financial Services—agreed to add the definition of eCheck to the X9.37 Electronic Check Exchange Standard.

Exhibit 2
eCheck Processing: The Four-Corner Model

NOTE: Model used when the first eCheck was issued in June 1998



1. Payer (the U. S. Treasury) sends a digitally-signed eCheck (including certificates representing the payer and their bank), along with payment advice data such as the supplier's (the payee's) invoice number and amount paid.

2a. Payee's (DoD suppliers such as GTE) eCheck processing system strips off the payment advice data and forwards it to its internal accounting system.

2b. Payee's eCheck processing system indorses the eCheck (by digitally signing the eCheck using the payee's certificate issued to them by the bank), digitally signs the entire message and forwards it to their bank (either BankBoston or NationsBank).

3a. Payee bank's eCheck processing system sends the deposit to the Demand Deposit Acount (DDA) system where the payee's account is credited and from which the payee is notified on their next statement.

3b. Payee bank's eCheck processing system encapsulates the eCheck into X9.46 formatted records (Financial Image Exchange) and sends it to the Fed for clearing.

4a. The Fed encloses the eCheck into an X9.37 format (Electronic Cash letter/Electronic Check Exchange) for clearing via Electronic Check Presentment (ECP) system. Being handled now like an electronic representation of a paper check, the Fed credits the payee bank's account and debits the payer bank's account. The eCheck is then forwarded to the payer's bank (in the pilot, also the Fed).

4b. Acting now as the Treasury's bank, the Fed debits the Treasury's account and sends a file of payments to the Treasury where they reconcile the eChecks that they wrote with those that cleared at the Fed.

Exhibit 3
Paper Check Processing Model

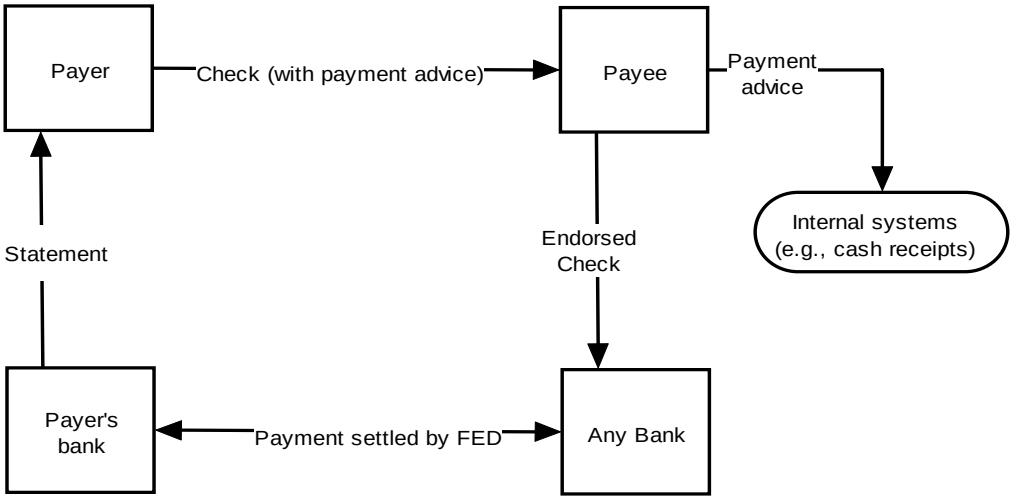
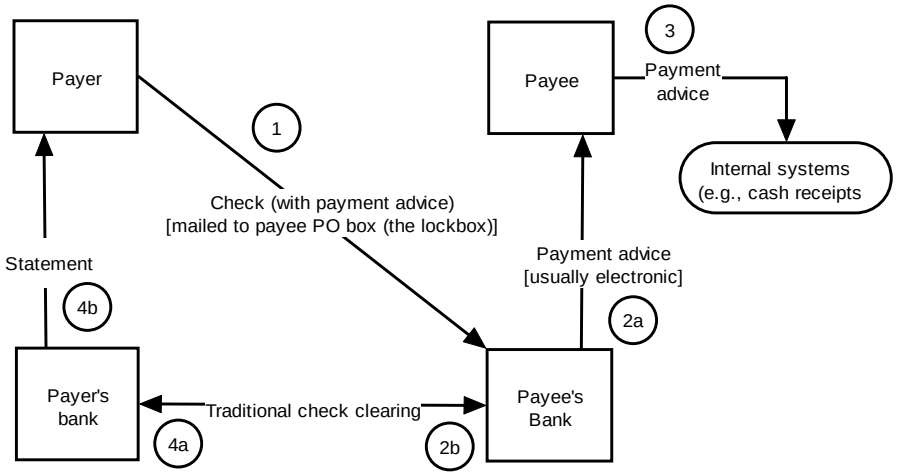


Exhibit 4
Lockbox Processing Model



1. Payer sends a paper check along with a paper payment (remittance) advice, such as a payment stub from an invoice, to the payee's PO box (the lockbox processing center operated by the payee's bank). The lockbox processing may be performed by a value-added network (VAN).

2a. Payee bank enters the payment advice and check data into their computer system and forwards the payment advice data to the payee. Copies of the payment advice documentation might be sent to the payee via courier or scanned and faxed.

2b. Payee bank sends the check to the payer's bank for processing.

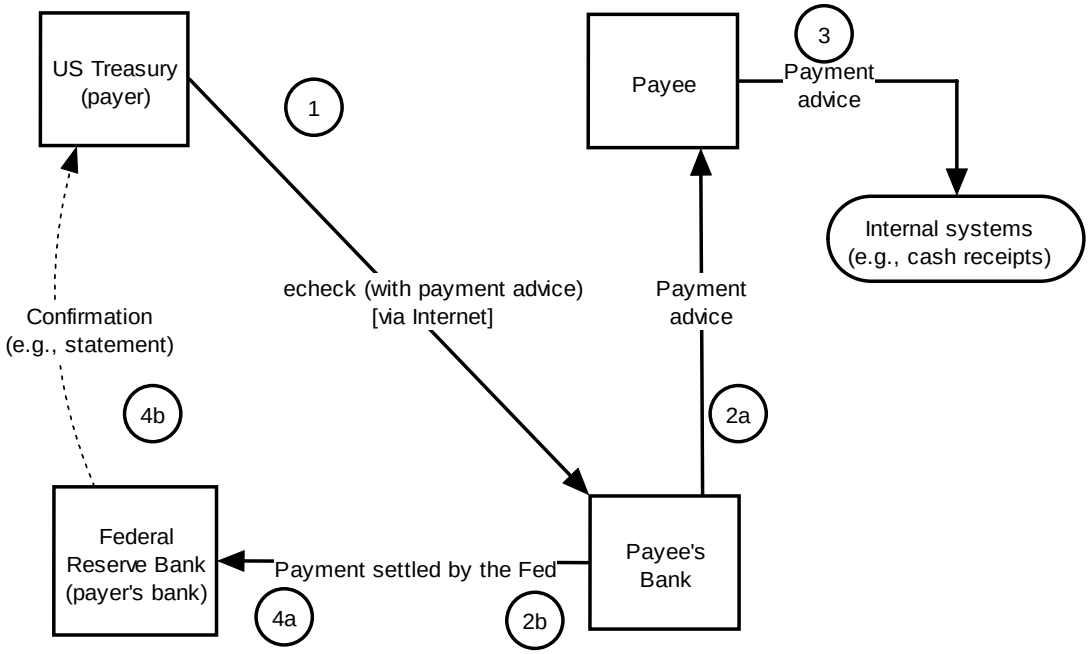
3. Payee sends the payment advice data to their internal accounting systems.

4a. The Fed processes the check by crediting the account for the payee's bank and debiting the account for the payer's bank.

4b. The payer's bank debits the payer's account and notifies the payer, via their next statement, that a deposit has been made

Exhibit 5
eCheck Processing: The Lockbox Model

Note: Proposed, but not used in the pilot



1. Payer (e.g., the U. S. Treasury) sends a digitally-signed eCheck (including certificates representing the payer and their bank), along with payment advice data such as the supplier's (the payee's) invoice number and amount paid to the electronic lockbox operated by the payee's bank.

2a. In the payee bank's (either BankBoston or NationsBank) electronic lockbox, the eCheck processing system strips off the payment advice data and forwards it to the payee.
The deposit data is sent to the Demand Deposit (DDA) system where the payee's account is credited.

2b. Payee bank's eCheck processing system prepares and sends to the Fed either an ACH debit, an eCheck for clearing through the traditional check clearing and settlement system, or a wire transfer.

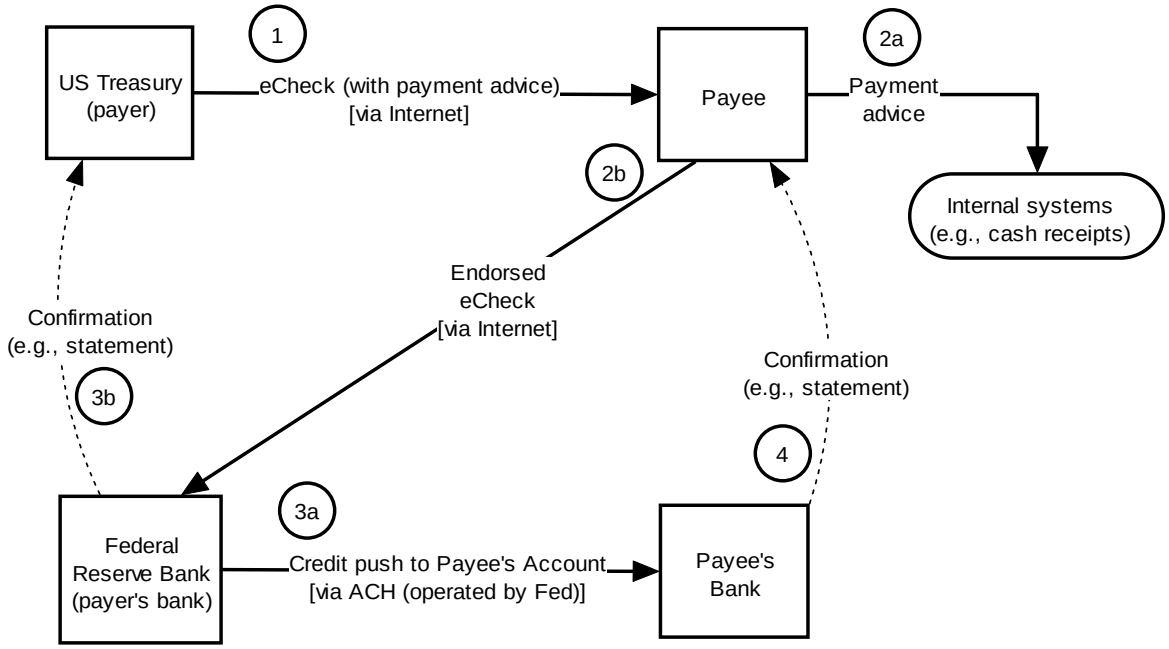
3. Payee sends the payment advice data to their internal accounting systems.

4a. The Fed processes the ACH debit, eCheck, or wire transfer by crediting the account for the payee's bank and debiting the account for the payer's bank. In this model the Fed is the payer's bank.

4b. Acting now as the Treasury's bank, the Fed debits the Treasury's account and sends a file of payments to the Treasury where they reconcile the eChecks that they wrote with those that cleared at the Fed.

Exhibit 6
eCheck Processing: The Z-flow Model Proposed by the Fed

Note: Proposed by Fed for phase 2 of the pilot



1. Payer (the U. S. Treasury) sends a digitally-signed eCheck (including certificates representing the payer and their bank), along with payment advice data such as the supplier's (the payee's) invoice number and amount paid.

2a. Payee's eCheck processing system strips off the payment advice data and forwards it to the payee's internal accounting system.

2b. Payee's eCheck processing system endorses the eCheck (by digitally signing the eCheck using the payee's certificate issued to them by their bank), digitally signs the entire message and forwards it to the payer's bank (the Fed).

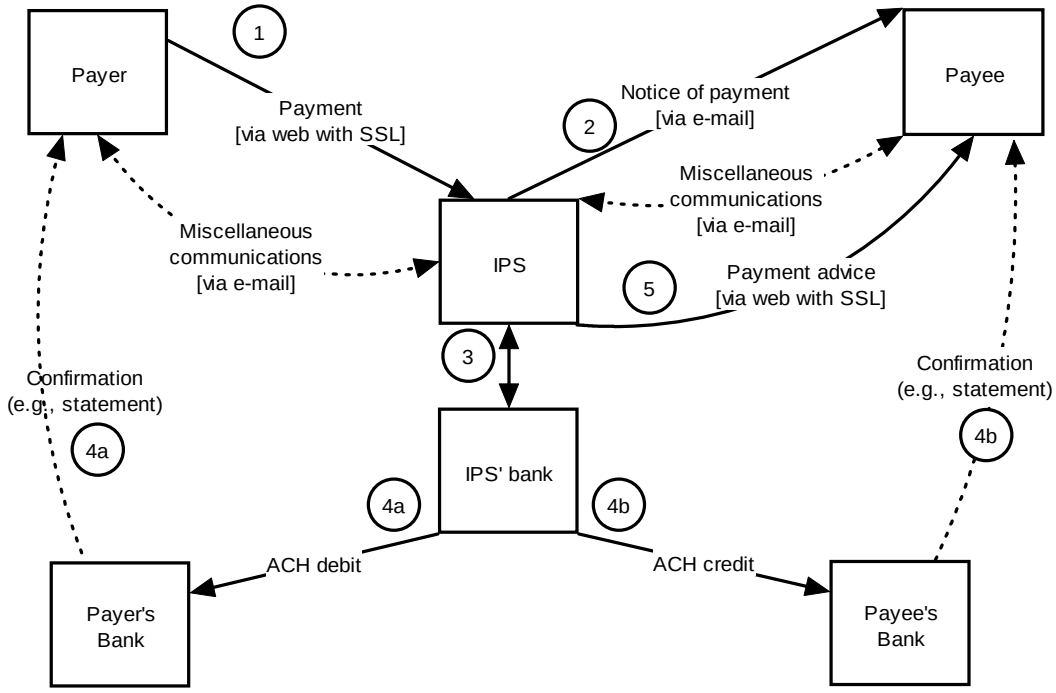
3a. The payer's bank (the Fed) originates an ACH credit that is sent to the payee's bank via the ACH network.

3b. Acting as Treasury's bank, the Fed debits Treasury's account and sends a file of payments to Treasury, which reconciles the eChecks that they wrote with those that cleared at the Fed.

4. Payee's bank credits the payee's account and notifies the payee on their next statement.

Exhibit 7
Mediated Flow Model Proposed by Frank Jaffe

NOTE: Internet Payment Service (IPS) Model proposed by Frank Jaffe for phase 2 of the pilot



1. Payer, using a Java applet that is delivered on demand by the IPS, makes a payment via the IPS web site. The session is secured by SSL and the payments are signed with a digital signature. The signing key is stored in an encrypted file on the payer's hard drive.

2. The IPS sends an e-mail to the payee notifying them of the payment.

3. The IPS notifies their bank of the payment.

4a. The IPS' bank generates an ACH debit. The Fed debits the payer bank's account at the Fed and sends the debit on to the payer's bank. The payer's bank debits the payer's account and notifies them via the regular bank statement.

4b. The IPS' bank generates an ACH credit. The Fed credits the payee bank's account at the Fed and sends the credit on to the payee's bank. The payee's bank credits the payee's account and notifies them via the regular bank statement.

5. The payee may obtain, via an SSL-secured web session, payment details that can be read directly into their accounting applications.

Information Systems & Computing
Academic Professionals

STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.

ISSN 1055-3096