1. Introduction to Electronic Checks

The Electronic Check was developed by a project of the Financial Services Technology Consortium. The FSTC is comprised of about 100 members, including most of the major banks, suppliers of technology to the financial industry, universities and research laboratories. The technical work of the Electronic Check Project was carried out in a number of phases: generating the original concepts, performing preliminary research, building and demonstrating a prototype, formulating specifications for a pilot system, and implementing the pilot system. Meanwhile the Project also carried out studies of the business, legal, regulatory, market and operational issues. Currently, the electronic check is being used in a pilot with the U.S. Treasury Department to pay suppliers to the Department of Defense.

The electronic check is designed to perform the payment and other financial functions of paper checks, by using cryptographic signatures and secure messaging over the Internet. The electronic check system is designed with message integrity, authentication and non-repudiation properties sufficient to prevent fraud against the banks and their customers. It is compatible with either interactive web transactions or with electronic mail. Since the electronic check does not depend on real-time interactions or on third party authorizations, electronic checks are better able to survive outages of network links and computing nodes. The result is a highly efficient electronic payments system, with a technology base that is extensible to a variety of financial instruments and other high-integrity document processing applications needed by the financial industry.

2. Executive Summary

A check is a signed paper document that orders the signer’s bank to pay an amount of money from the signer’s account after a specified date. Paper checks are the most widely used payment instrument after cash in the United States. They have the advantage that payers and payees can be individuals, small businesses, brokerages, corporations, governments or almost any other type of organization. They pass directly from the payer to the payee, so that the timing and the purpose of the payment is clear to the payee. While checks are usually very simple, business checks can require multiple makers signatures and can be accompanied by lists of invoices being paid. Checks can be made out to require a single endorsement or multiple endorsement. The payee can deposit a check in an account of his choice or cash it. Banks operate extensive facilities to accept checks for deposit, process them internally, and clear and settle between banks. Banks also microfilm checks for their archives and may return checks with customer account statements.

The electronic check, or echeck, is based on the idea that electronic documents can be substituted for paper and that public key cryptographic signatures can be substituted for handwritten signatures. Therefore, the echeck can replace paper checks without the need to create a new payment instrument, along with the legal, regulatory and commercial practice changes that a new
payment instrument would imply. Instead, the echeck is designed to fit into current check practices and systems with minimum impact on payers, payees, banks, and the financial system.

Figure 1 shows the normal flow of an echeck. The payer writes an echeck by structuring an electronic document with the information legally required to be in a check and cryptographically signs it. The payee receives the echeck, verifies the payer's signature, endorses the echeck, writes out a deposit, and signs the deposit. The payee's bank verifies the payer's and payee's signatures, credits the payee's account and forwards the check for clearing and settlement. The payer's bank verifies the payer's signature and debits the payer's account. Note that cryptographic signatures can be verified on every echeck, while handwritten signatures are rarely verified.

![Figure 1. Electronic Check Basic Flow](image)

Since an echeck must contain specific mandatory and optional information and cryptographic signatures, the echeck is written in FSML (Financial Services Markup Language), a language specified using SGML (Standard Generalized Markup Language). SGML is a standard language for specifying markup languages. The document structure and data items for echecks are delimited by “tags” similar to those used in HTML (HyperText Markup Language), another language defined using SGML.

FSML is designed to support the data structures and cryptographic signatures needed for echecks, but it can be generalized and extended to other financial services documents.

The echeck written in FSML will contain all of the information that is normally found in paper checks, including that which is handwritten, preprinted and in the magnetic ink character line at
the bottom of the check. In particular, it will contain the information needed to interface with existing bank DDA (Demand Deposit Account) systems for posting of echecks and deposits.

FSML includes signature blocks designed to support the addition and deletion of FSML document blocks, and to support rich signature semantics to enable signing, co-signing, and endorsing of echecks by the several parties who may sequentially process an echeck. The FSML structure and signature mechanisms also provide the ability to encapsulate and cryptographically attach other documents, such as an advice of payment, invoice, or remittance information to enable the payee to correctly post the payment in accounts receivable.

The FSML signature mechanisms have been documented separately as SDML (Signed Document Markup Language), which is general enough to support secure messaging applications which require authenticated signatures of multiple parties on electronic documents that undergo additions and deletions during their life.

X.509 certificates are used with echecks to provide the verifiers of public key signatures with the signer’s public signature verification key. The bank issues a certificate when the bank customer opens an electronic checking account, and it will reissue the certificate upon expiration, provided that the account is in good standing and that no theft or misuse of the signer’s private signature key has been reported. The X.509 certificate only informs the signature verifier that the public key was legitimately associated with a signer and bank account at the time that the certificate was issued. The X.509 certificate does not imply that the echeck is guaranteed in any sense. For example, there may not be sufficient funds in the account it is drawn on. Since only the paying bank can know both the current status of the account and the most recent reports of private signature key theft and misuse, only the paying bank can decide whether the echeck is genuine and payable. Other verifiers of echeck signatures can trust that the check was signed using a private signing key that belonged to a legitimate echeck account holder and that the echeck has not been altered. Since certificate revocation lists would only give information about key status and not account status, their publication is not planned.

An electronic checkbook smart card is used to protect the signer’s private signature key from theft and misuse. The use of cryptographic hardware gives a signature verifier more confidence that the echeck came from the legitimate account holder because the private signing key is generated and used only within the smart card by cryptographic algorithms meeting banking industry standards. The private signing key is never transferred to the signer’s computer, and it is never exposed to theft via the computer’s network connection. The electronic checkbook also automatically numbers each check when it is signed in order to ensure echeck uniqueness, and it keeps a log or check register to be consulted in case there is a dispute as to whether a given echeck has been signed, endorsed, or deposited. Use of the electronic checkbook is controlled by entry of a PIN by the signer.

To ensure the widest possible compatibility with electronic mail, HTTP (HyperText Transport Protocol) and other types of transport, FSML specifies a limited line length and a restricted ASCII character set for encoding all echeck data.

The cryptographic signatures are sufficient in the echeck system to secure the echeck against fraud by ensuring message integrity, authentication and non-repudiation. Thus, the echeck system
and the application-level cryptography can be exported and used internationally. When confidentiality is needed between any two parties, encryption can be used on that transport link. For example, encrypted email may be used between the payer and payee or between the payee and bank.

Echecks are currently cleared and settled between the banks using the ANSI X9.46 and X9.37 standards. The echeck FSML representation is carried in X9.46 in place of a check image. The information that would be in a paper check MICR line is copied into an X9.37 cash letter which is also carried within the X9.46 format.

The Electronic Check Clearing House Organization (ECCHO) has adopted rules for inter-bank clearing of electronic checks which consider the echeck to have the status of a "negotiable instrument" for the purposes of the Uniform Commercial Code and a "check" for purposes of Regulation CC. This is reinforced by including a legal notice within the echeck which states "This instrument subject to check law", as well as by uniform customer agreements for use between the banks and their customers, which agree that echecks are subject to check law. If a consumer, rather than a business, writes the electronic check, then Regulation E also applies. Regulation E provides consumers with additional rights and protections, and it supercedes Regulation CC where they overlap.

Figure 2. Electronic Check Concept

In the normal check flow described in this overview, the banks providing echecking accounts must be members of ECCHO, members of an equivalent organization or clearing house providing clearing and settlement rules, or they can agree to clear and settle echecks bilaterally. An exception to this is the alternative echeck flow described in Figure 4 below, where other
networks, such as Automated Clearing House, can be used to transfer funds between the banks. While it would be an obvious extension of the echeck technology for the payee’s bank to collect the funds from thepayer’s bank by generating an ACH debit, this extension would involve legal and regulatory considerations that go beyond checking. Truncation of echecks, either by use of ACH or by means that parallel the truncation of paper checks, is a topic of further study.

3. Electronic Check Concepts

3.1. Operational Characteristics

In the example shown in Figure 2, the business transaction begins with the payee sending an invoice or bill to the payer, which is processed by the payer’s accounts payable system. When the time comes to pay the invoice, the invoice information is retrieved from the accounts payable system, and the invoice data is used to create an echeck. The echeck includes familiar check information such as the payee’s name, the amount, and the date and the account information. To sign the echeck, the payer enters a PIN to unlock an electronic checkbook card in the form of a smart card. This card is a secure container for the payer’s private signature key, and assures a degree of non-repudiation. The signature on the echeck may also cryptographically bind a copy of the invoice to the echeck, so that an attacker cannot substitute a different invoice in order to commit fraud. The invoice format is not fixed, but it can be flexible with respect to length, format and data content, so that the payer can return the document received from the payee. This provides the payee with the complete information needed to correctly post the payment.
The signed echeck and invoice is sent to the payee by email or a web transaction. The payee verifies the payer's signature on the echeck and invoice, detaches the invoice information, and posts the payment to accounts receivable. The payee enters his PIN to unlock his electronic checkbook and uses the electronic checkbook to endorse the echeck and to sign an electronic deposit slip to deposit a batch of echecks.

The endorsed echeck is forwarded to the payee’s bank for deposit and subsequent clearing. The clearing process can be done by integrating echeck into existing Electronic Check Presentment systems or other clearing and settlement systems. Both the payee’s bank and payer’s bank verify all signatures on the echeck and endorsement using a two layer certificate system which links the signature verification keys to the signer and signer's bank account. The paying bank verifies that this transmission of the echeck is not a duplicate, that the payer's certificate and account are currently valid, and posts the echeck to the payer's Demand Deposit Account (DDA).

Finally, the payer receives a line item on his statement, which may now carry a full description of the transaction, since the entire contents of the echeck are machine-readable.

Echecks have been designed so that the integrity, authentication and non-repudiation properties of public key signatures are sufficient to protect against fraud. Furthermore, to protect the paper check accounts against fraud, echecks use different bank account numbers, which are valid only for cryptographically signed echecks. Since encryption is not required to prevent fraud, the echeck technology is compatible with export policies regarding encryption technology. The echeck may be encrypted over any of the transmission links for privacy reasons, using encryption technology of a type and strength consistent with regulations governing each situation.
Furthermore, the payer and payee are not anonymous to their respective banks, and echecks are compatible with legal requirements to report certain types of financial transactions.

3.2. Alternative Operational Models

The echeck flow shown in Figure 2 is typical, but echecks can be used in other ways, just like paper checks.

Figure 3 shows how a lockbox operator can process an echeck on behalf of the payee. In this case the lockbox operator does the cryptographic processing to verify the payer’s signature. The invoice or advice of payment information can be converted by the lockbox operator to the same format used for paper checks. This allows the payee, typically a biller, to receive echeck payments without implementing new, echeck-specific software and hardware. The lockbox operator endorses and deposits the echeck on behalf of the payee. If the lockbox is operated by the payee’s bank, then the lockbox functions can be interfaced directly to the bank’s echeck server, saving the separate endorsement and deposit signing steps.

Figure 5. Certified Electronic Check Flow

Figure 4 shows how a payee can endorse and cash the echeck at the payer’s bank. The payer’s bank transfers the proceeds to the payee’s bank by electronic funds transfer. This flow is particularly useful if a payer wishes to pay payees whose bank does not yet offer echeck accounts. The payer’s bank can provide the payee with a “check cashing” electronic checkbook valid for cashing the payer’s checks. The electronic funds transfer can be done over a variety of networks, including international networks. In this case the payer could write the echeck in the payee’s currency, and the payer’s bank could implement the funds conversion from the payer’s currency to the payee’s currency when making the electronic funds transfer.
Figure 5 shows a certified electronic check flow. In this case, the payer sends the echeck to the payer's bank. The payer's bank verifies the payer's signature, determines that there are funds available in the payer's account, and places a hold on the funds. The payer's bank then countersigns the check to certify it, and send the check back to the payer. Alternatively, the bank can send the certified check directly to the payee, possibly over an encrypted channel to provide the payee with the greatest degree of security and confidentiality.

3.3. **Electronic Check Truncation**

The ECCHO Electronic Check Operating Rules define presentment as occurring when the echeck arrives at a location specified by the paying bank. While echecks require significantly fewer bytes per item than do images of paper checks, particularly if the X.509 certificates are passed by reference instead of value and if lossless compression is used, additional cost reductions can be achieved by truncating echecks at the bank of first deposit. It is expected that echeck truncation will take advantage of technical and business arrangements being developed for truncation of paper checks and check images.

3.4. **Financial Services Markup Language**

The Financial Services Markup Language (FSML) is defined using ISO 8879 Standard Generalized Markup Language (SGML), supplemented by additional requirements. An SGML Document Type Definition specifies how tags are used to structure the document as well as to identify the individual data items. A wide variety of tools exist to process SGML documents, and this makes it easy for developers of electronic commerce software to integrate echecks into their products.

7-bit ASCII characters strings are used to encode the values of integer, decimal and character data items, with hex strings being used to encode large binary objects such as hash values and X.509 certificates. These encodings are designed to be compatible with Internet SMTP mail and any other commonly used transport system capable of moving 7-bit text, such as web forms. Although less compact than binary encodings, the ASCII encodings are readily compressed by modems or by other standard compression algorithms for bulk transmission or archival storage. A humanly readable format is also easier to debug, both in development and when problems occur in installation and production.

Each FSML Document is a sequence of blocks.1 The start tag and end tags identify the type, and hence the contents, of each block. A good example is the check block shown in Table 1. For all types of blocks, the start tag is always followed by the block name, criticality and version number data items. In some block types, such as check, the first part of the block content will be a

---

1 An FSML document can contain blocks and embedded FSML documents. The description of FSML and the included blocks is intended to provide an introduction to the principle concepts of FSML. For a full description of FSML see References FSML Version 1.17 and SDML Version 1.0.
sequence of data items which are logged by the electronic checkbook. This gives the signer a secure, portable record of the most important items which have been signed by the private signing key in the electronic checkbook.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Data</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;check&gt;</td>
<td></td>
<td>Start tag of the check block</td>
</tr>
<tr>
<td>&lt;blkname&gt;</td>
<td>string</td>
<td>Name by which the block is referenced</td>
</tr>
<tr>
<td>&lt;crit&gt;</td>
<td>&quot;true&quot;</td>
<td>Check is a critical block</td>
</tr>
<tr>
<td>&lt;vers&gt;</td>
<td>decimal</td>
<td>Version number of this block</td>
</tr>
<tr>
<td>&lt;checkdata&gt;</td>
<td></td>
<td>Start tag of data elements (shaded area below) to be logged in electronic checkbook register</td>
</tr>
<tr>
<td>&lt;checknum&gt;</td>
<td>integer</td>
<td>Check number automatically incremented by electronic checkbook</td>
</tr>
<tr>
<td>&lt;dateissued&gt;</td>
<td>ISO date</td>
<td>Year, month, day that check was written</td>
</tr>
<tr>
<td>&lt;datevalid&gt;</td>
<td>ISO date</td>
<td>Year, month, day that check is payable</td>
</tr>
<tr>
<td>&lt;country&gt;</td>
<td>ISO country</td>
<td>Optional, check is subject to this country's law</td>
</tr>
<tr>
<td>&lt;amount&gt;</td>
<td>decimal</td>
<td>Numerical amount of currency</td>
</tr>
<tr>
<td>&lt;currency&gt;</td>
<td>ISO currency</td>
<td>3 character currency code, e.g. USD for US dollars</td>
</tr>
<tr>
<td>&lt;payto&gt;</td>
<td>string</td>
<td>Name of person to whose order the check is payable, may repeat for joint or several payees</td>
</tr>
<tr>
<td>&lt;paytobank&gt;</td>
<td>string</td>
<td>Optional, bank routing code of payee's bank</td>
</tr>
<tr>
<td>&lt;paytoaccount&gt;</td>
<td>string</td>
<td>Optional, payee's account at payee's bank</td>
</tr>
<tr>
<td>&lt;paytocustno&gt;</td>
<td>string</td>
<td>Optional, payee's unique identifier at payee's bank</td>
</tr>
<tr>
<td>&lt;/checkdata&gt;</td>
<td></td>
<td>End tag of check data elements (shaded area above) to be logged in electronic checkbook register</td>
</tr>
<tr>
<td>&lt;checkbook&gt;</td>
<td>integer</td>
<td>Manufacture, model, serial number of payer's electronic checkbook</td>
</tr>
<tr>
<td>&lt;restrictions&gt;</td>
<td>string</td>
<td>Optional, &quot;duration pynmnd&quot;, &quot;for deposit only&quot; or &quot;all payees must endorse&quot; (pynmnd is an ISO standard format for representing a period of time as n years, n months and n days).</td>
</tr>
<tr>
<td>&lt;paytokey&gt;</td>
<td>hexstring</td>
<td>Optional, public key to be used to verify payee's endorsement</td>
</tr>
<tr>
<td>&lt;payeracct&gt;</td>
<td>string</td>
<td>Optional, payer's account with payee's business</td>
</tr>
<tr>
<td>&lt;memo&gt;</td>
<td>string</td>
<td>Optional</td>
</tr>
<tr>
<td>&lt;conditions&gt;</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td>&lt;legalnotice&gt;</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td>&lt;/check&gt;</td>
<td></td>
<td>End tag of check block</td>
</tr>
</tbody>
</table>

Table 1. Check Block

The number of the electronic checkbook and the check number are automatically inserted by the electronic checkbook to ensure uniqueness of each echeck. The check block requires the date that the echeck becomes valid and the date that the echeck was issued. The writer may enter the
country that the echeck was written in order to comply with local laws. The writer must specify the amount and type of currency, and the “pay to the order of” data item must be present. However, in order to unambiguously identify the payee, the payer may specify the payer’s bank and account or customer number at the bank. Alternatively the payer may specify the payee’s public key. A payer may place restrictions on the echeck, such as to distinguish between writing an echeck to “John and Mary Smith”, which both must endorse, or an echeck to “John or Mary Smith” which either can endorse. Just as on paper checks, the payer may enter memo information as a reminder of the purpose of the echeck. A specific memo field is provided which the payer may use to enter the payer’s account number at the payee’s business to assist the payee in correctly posting the payment in the payee’s accounts receivables. Lastly, the check block carries a legal notice that the echeck is subject to check law.

<table>
<thead>
<tr>
<th>Tag*</th>
<th>Data</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;account&gt;</td>
<td>Start tag of the account block</td>
<td></td>
</tr>
<tr>
<td>&lt;blkname&gt;</td>
<td>string</td>
<td>Name by which the block is referenced</td>
</tr>
<tr>
<td>&lt;crit&gt;</td>
<td>&quot;true&quot;</td>
<td>Account is a critical block</td>
</tr>
<tr>
<td>&lt;vers&gt;</td>
<td>decimal</td>
<td>Version number of this block</td>
</tr>
<tr>
<td>&lt;bankcode&gt;</td>
<td>string</td>
<td>Bank routing code of the issuing bank</td>
</tr>
<tr>
<td>&lt;bankacct&gt;</td>
<td>string</td>
<td>Account number of this account at the issuing bank</td>
</tr>
<tr>
<td>&lt;bankser&gt;</td>
<td>integer</td>
<td>Serial number of this account block, assigned by the issuing bank</td>
</tr>
<tr>
<td>&lt;expdate&gt;</td>
<td>ISO date</td>
<td>Expiration date of this account block</td>
</tr>
<tr>
<td>&lt;accttitle&gt;</td>
<td>string</td>
<td>Optional, account title</td>
</tr>
<tr>
<td>&lt;accttype&gt;</td>
<td>string</td>
<td>Optional, account type</td>
</tr>
<tr>
<td>&lt;bankname&gt;</td>
<td>string</td>
<td>Optional, name of the issuing bank</td>
</tr>
<tr>
<td>&lt;bankaddr&gt;</td>
<td>string</td>
<td>Optional, address of the issuing bank</td>
</tr>
<tr>
<td>&lt;acctrest&gt;</td>
<td>string</td>
<td>Optional, a string containing &quot;minimum amount nnnnn.nn ccc&quot;, &quot;maximum amount nnnnn.nn ccc&quot;, &quot;n signatures required&quot;, &quot;n signatures required above amount nnnnn.nn ccc&quot;, &quot;special processing&quot;, or &quot;currency ccc&quot;.</td>
</tr>
<tr>
<td>&lt;certissuer&gt;</td>
<td>string</td>
<td>The unique distinguished name of the issuer of the certificate that can be used to verify signatures on FSML documents containing this account block.</td>
</tr>
<tr>
<td>&lt;certserial&gt;</td>
<td>integer</td>
<td>The unique certificate serial number of the certificate.</td>
</tr>
<tr>
<td>&lt;/account&gt;</td>
<td>End tag of account block</td>
<td></td>
</tr>
</tbody>
</table>

* Some optional tags have been omitted for clarity.

Table 2. Account Block.

Another block that is part of every echeck is the account block shown in Table 2. The account block contains the bank code and account number that correspond to the bank code and account number that would be found on the MICR line of a check, and it contains a reference by issuer and serial number to a certificate containing a public key that can be used to verify signatures on echecks drawn against the account. It may also contain restrictions on the value, the maximum
time interval before the echeck becomes stale, and requirements on the number of signatures required for echecks above certain values.

A bank customer may have more than one account block corresponding to the several accounts over which the customer has signature authority. There also will be multiple account blocks issued on a single account when the account is held jointly or in the case of commercial accounts where several persons have signature privileges.

Block types which are currently defined for echecks are shown in Table 3.

<table>
<thead>
<tr>
<th>Block Type</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;account&gt;</td>
<td>A block containing information about the payer's or payee's account, such as the bank code, account code, account title, signer’s name and address, and the issuer and serial number of the signer’s certificate that may be used to verify signatures on echecks drawn against the account. The account block is signed by the bank holding the account.</td>
</tr>
<tr>
<td>&lt;action&gt;</td>
<td>A block describing the action to be performed by the recipient.</td>
</tr>
<tr>
<td>&lt;attachment&gt;</td>
<td>A block containing information sent from the payer to payee, and which is to be associated with the echeck.</td>
</tr>
<tr>
<td>&lt;bankstamp&gt;</td>
<td>A block containing processing status information.</td>
</tr>
<tr>
<td>&lt;bundle&gt;</td>
<td>A block containing totals for a bundle of echecks being sent between banks.</td>
</tr>
<tr>
<td>&lt;cashletter&gt;</td>
<td>A block containing cash letter totals for echecks being sent between banks.</td>
</tr>
<tr>
<td>&lt;cert&gt;</td>
<td>A block containing an X.509 certificate. Certificates are needed for bank customers, bank tellers, bank information systems, and certificate management purposes.</td>
</tr>
<tr>
<td>&lt;certification&gt;</td>
<td>A block containing information provided by a bank, along with the bank's signature, to create a certified echeck.</td>
</tr>
<tr>
<td>&lt;check&gt;</td>
<td>A block containing echeck information such as the amount, pay-to-the-order-of, and the date on which the echeck becomes payable.</td>
</tr>
<tr>
<td>&lt;deposit&gt;</td>
<td>A block containing information about a set of echecks being deposited as a batch.</td>
</tr>
<tr>
<td>&lt;endorsement&gt;</td>
<td>A block containing information supplied by the endorser, such as date, “for deposit only”, etc.</td>
</tr>
<tr>
<td>&lt;invoice&gt;</td>
<td>An invoice/remittance document containing payment information sent between payer and payee.</td>
</tr>
<tr>
<td>&lt;signature&gt;</td>
<td>A block containing a list of the names of other blocks being signed, a list of their hashes, a reference to the public key certificate needed to verify the signature, and a cryptographic signature on the above.</td>
</tr>
</tbody>
</table>

Table 3. FSML Block Types Defined for Echecks

3.5. Cryptographic Signatures

When an echeck is originated, a minimum set of information is written and signed. As it is processed, more information and more signatures will be added as it is passed from party to party. For example, an echeck may be:
- originated by a payer,
- co-signed by a co-payer,
- certified by a bank,
- endorsed by a payee,
- co-endorsed by a co-payee,
- deposited, and
- paid.

<table>
<thead>
<tr>
<th>Tag*</th>
<th>Data</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;signature&gt;</td>
<td></td>
<td>Start tag of the signature block</td>
</tr>
<tr>
<td>&lt;blkname&gt;</td>
<td>string</td>
<td>Name by which the block is referenced</td>
</tr>
<tr>
<td>&lt;crit&gt;</td>
<td>&quot;true&quot;</td>
<td>Signature is a critical block</td>
</tr>
<tr>
<td>&lt;vers&gt;</td>
<td>decimal</td>
<td>Version number of this block</td>
</tr>
<tr>
<td>&lt;sigdata&gt;</td>
<td></td>
<td>Start tag of data in this block which is hashed and signed by the cryptographic signature at the end of the block</td>
</tr>
<tr>
<td>&lt;blockref&gt;</td>
<td>string</td>
<td>Name of the block whose hash is next</td>
</tr>
<tr>
<td>&lt;hash alg=&quot;TYPE&quot;&gt;</td>
<td>hexstring</td>
<td>Hash of the block just named</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The TYPE attribute identifies the hash algorithm used to hash the block, e.g. &quot;sha&quot; for NIST Secure Hash Algorithm</td>
</tr>
<tr>
<td>&lt;nonce&gt;</td>
<td>string</td>
<td>A random string generated by the electronic checkbook and used to render the blocks unpredictable to attackers before they are hashed</td>
</tr>
<tr>
<td>&lt;sigref&gt;</td>
<td>string</td>
<td>Optional, the name of an account block or cert block containing a reference to the certificate or the certificate needed to verify this signature</td>
</tr>
<tr>
<td>&lt;certissuer&gt;</td>
<td>string</td>
<td>Optional, the unique distinguished name of the issuer of the certificate that can be used to verify signatures on FSML documents containing this account block.</td>
</tr>
<tr>
<td>&lt;certserial&gt;</td>
<td>integer</td>
<td>Optional, the unique certificate serial number of the certificate.</td>
</tr>
<tr>
<td>&lt;algorithm&gt;</td>
<td>string</td>
<td>Hashing and signing algorithms used to compute the signature on &lt;sigdata&gt;, either &quot;md5/rsa&quot;, &quot;sha/dsa&quot;, or &quot;sha/rsa&quot;</td>
</tr>
<tr>
<td>&lt;username&gt;</td>
<td>string</td>
<td>Optional, the signer's name</td>
</tr>
<tr>
<td>&lt;userphone&gt;</td>
<td>string</td>
<td>Optional, the singer's phone number</td>
</tr>
<tr>
<td>&lt;/sigdata&gt;</td>
<td></td>
<td>End tag of data in this block which is hashed and signed by the cryptographic signature at the end of the block</td>
</tr>
<tr>
<td>&lt;sig&gt;</td>
<td>hexstring</td>
<td>Cryptographic signature computed by the electronic checkbook using the signer's private key</td>
</tr>
<tr>
<td>&lt;/signature&gt;</td>
<td></td>
<td>End tag of signature block</td>
</tr>
</tbody>
</table>

* Some optional tags have been omitted for clarity.

Table 4. FSML Signature Block

Some of the additional information, such as the certification or endorsement information, are permanently part of the echeck and remain intact until it is returned to the payer. Other
information, such as a remittance slip, may be associated with the echeck for part of its life (e.g. transmittal from payer to biller), but are then removed and processed separately.

This requires a flexible document structure and signature mechanism. The principle characteristics of the FSML signature mechanisms are:

- The document consists of a sequence of blocks, and blocks may be nested.
- Signatures implement cryptographic signatures and/or hashes on other blocks referred to by name.
- Signatures may sign other signature blocks.
- Signature blocks refer by block name or by issuer and serial number to the certificate block which contains the corresponding public signature verification key.

These characteristics provide enough flexibility to handle the signature requirements of electronic checks and the signature requirements of a broad range of financial documents.

The signature block contains the names of the other blocks being signed, and the corresponding hashes computed on those blocks. The <blockref> and <hash> data elements are repeated in pairs (as shaded) when multiple blocks are being signed. The random nonce is generated by the electronic checkbook prior to hashing any of the blocks, it is prepended to each block, and it is also included in the scope of the hash calculation on the signature block itself. This defeats attacks where the attacker fabricates two messages that have the same hash value, persuades the victim to sign one, and then substitutes the other.

The signature block also contains a reference to the public key to be used in verifying the signature, either by naming another FSML block or by referencing the X.509 certificate by the Certificate Authority’s distinguished name and the certificate serial number.

The signer can choose to include other personal data, such as name, address, phone number, email address, etc. These data items are stored in the electronic checkbook when it is initialized by the bank, and are changeable only after the electronic checkbook has been unlocked using the bank's administrative PIN. This method of providing personal information is not as secure as including the information in the X.509 certificate or the account block. However, it is expected to be secure enough for purposes such as analysis and decision making by check guarantee services, while providing the bank’s customer with complete control over which personal information is released to which payee.

The electronic checkbook signature is actually a signature on a hash of hashes. The advantage of this design is that the signature can be verified for any subset of the blocks which were originally signed. However, it also implies that if any blocks are absent, the application can determine whether the subset of blocks still present constitute a valid echeck. For example, attachments which might contain remittance or other business information are deemed not to affect the validity of the echeck, and attachment blocks can be removed by the payee or discarded by the bank of first deposit with impunity.

Figure 6 shows how signatures are applied to the echecks. The right hand set of blocks are the check as made out by the payer. The payer's signature signs the action, check, account, attachment and invoice blocks. The payer's signature block references the payer's account block.
which references the payer's certificate. The payer's certificate contains the payer's public key, which verifiers use to verify the payer's signature. The payer's bank has signed both the payer's account and certificate blocks, and the payer's bank's certificate contains the public key needed to verify the bank's signature.

Figure 6. Electronic Check Signatures and Endorsement

The endorser has signed the action and endorsement blocks of the endorsement, plus the check and the payer's signature on the echeck being endorsed.

The depositor has signed the action and deposit blocks of the deposit, plus the endorsement and endorser's signature of the echeck being deposited. Note that since the payer's signature block only contains the hashes of the attachment and invoice blocks, when these blocks are removed by the depositor, the payer's signature can still be verified on the remaining check blocks by the paying bank.

3.6. Electronic Checkbooks

A handwritten signature captures the reflexive movement of the signer's muscles and is partly a biometric characteristic of the signer. This makes it difficult for a forger to create a perfect forgery, even if the forger has a sample of a handwritten signature.

However, a perfect forgery of a cryptographic signature can be made by anyone who has the signer's private signature key.
Therefore, it is critical to establishing an echeck system based on public key signatures that payees and banks are able to trust that payers can maintain possession and control over their private signing keys at all times. Electronic checkbook smart cards or other cryptographic hardware are used to help ensure that signatures are made only by legitimate signers, so that check forgery is made difficult. Electronic checkbook cards also standardize and simplify key generation, distribution and use, so that a high and uniform level of trust can be established without depending on the skill and diligence of every echeck customer.

Figure 7. Public Key Signature Security Fundamentals

As shown in Figure 7, verifiers must believe three types of things about the signer and the signer's private and public signing keys:

1. **Private key possession and control** -- The signature verifier must believe that the signer has exclusive possession of his signing key. If an attacker can get possession of the signer's private key, then the attacker can forge signatures using his own system from any location on the network. If the attacker can gain control of the signer's computer, then the attacker can forge signatures without the signer's knowledge.

The electronic checkbook, in the form of a PIN-activated tamper-resistant smart card or similar cryptographic hardware, performs the signing algorithm so that the private signing key is always kept inside the trusted hardware and is never read into the signer's much more vulnerable networked personal computer or server. The electronic checkbook is aware of echeck syntax and logs critical data from echecks to provide the signer with a trusted log of signing actions.
2. **Key pair generation** -- The signature verifier must believe that the private/public key pair was generated such that the private key cannot be calculated or guessed by an attacker based on knowledge of the public key.

   The electronic checkbook performs key generation within the tamper-resistant hardware using algorithms that have been properly tested and certified by the manufacturer. Only the public key is exported from the hardware, and the private key is never revealed to anyone.

3. **Public key infrastructure** -- The signature verifier must be able to trust that the public key provided for use in verifying the signature really belongs to the signer and is the other half of the signer's public key pair.

   The electronic checkbook is initialized using procedures based on bank card issuing processes. The public key exported from the card is included in an X.509 certificate signed by the bank's Certification Authority, and associated with an account block also signed by the bank's Certification Authority. The bank echeck servers also keep an independent database of the bank's signer's public keys, such that they always know the most current relationships of keys to accounts and signers.

   Besides key management and logging functions just described, electronic checkbooks also:
   1. include the checkbook's unique number (manufacturer, model and serial number) in each echeck,
   2. consecutively number each echeck as it is signed, in order to ensure that each echeck is unique,
   3. generate random numbers that are prefixed to blocks to increase security of the hashing functions,
   4. contain signer personal data which the signer can selectively apply to echecks,
   5. separately unlock echeck writing, checkbook administration and bank administration functions using PINs,
   6. deactivate if PIN hacking is detected.

   Furthermore, the design of the electronic checkbook must be such that the private signing key cannot be extracted via its electrical connector and any successful attempt to extract the private key will visibly damage the electronic checkbook and render it inoperable.

### 3.7. **Fraud Prevention**

A design goal of the echeck system is to prevent fraud without relying on encryption, since standard widespread availability of strong encryption has been hampered by export controls and by attempts to regulate its use. Cryptographic signature systems are quite freely used and exported, while key management and public key infrastructure components are more restricted, especially if they can also be used to manage encryption keys.

Features intended to prevent fraud include:
1. **Duplicate detection** -- Each echeck is guaranteed to be unique by the operation of the electronic checkbook. The payee and payee's bank should detect and refuse duplicate submissions of echecks, and the payer's bank must detect duplicates and pay only one instance of an echeck. This prevents multiple payments due to innocent retransmissions of email, and also prevents a payee from cashing and depositing an echeck in two different accounts.

2. **Payee identification** -- The check block provides for checks to be made out to the payee's bank routing code and either an account or customer ID number. It also provides for an echeck to be made out to the payee's public key. These parameters uniquely identify the payee, and prevent an eavesdropper from exploiting the ambiguity of payee identification which otherwise exists if only payee common names are used. These parameters are also included in the endorsement block so that echecks can be endorsed over to uniquely identified third parties.

3. **Electronic account numbers** -- The account number in the account block is a randomly chosen number assigned by the bank for the purpose of writing and depositing echecks. The payer's and depositor's echeck account numbers are mapped to their paper check account numbers by their respective banks. The banks will not accept paper checks or drafts written against the echeck account numbers. This prevents an eavesdropper or corrupt payee from printing and passing paper checks or drafts against the account numbers.

4. **Cryptographically attached invoices** -- Invoice and attachment blocks can be sent with the echeck blocks to detail the purpose of the payment. These blocks can be signed by the echeck signature, which binds these documents to the echeck and ensures their authenticity and integrity. This prevents an attacker from, for example, intercepting an echeck and purchase order, changing the delivery address in the order, and forwarding the echeck and altered order to the merchant.

3.8. **Privacy**

Although encryption is not required to protect echecks against fraud, encryption will be desired in many cases between payer and payee or between payee and the payee's bank in order to protect the privacy of the transaction. Echecks may be transported in HTTP over SSL or sent as MIME parts in S/MIME. The key management and public key infrastructures for these systems will be different from that for echeck signature keys, and the keys are typically stored in files on the user's personal computer. Users are able to use 40-bit or 128-bit keys as the situation dictates, or to employ encryption hardware depending on their specific privacy needs.

4. **Payer and Payee Systems**

Payer and payee systems are used to write echecks, verify signatures, endorse echecks and prepare deposits for transmission to the banks. They also provide network interfaces and may provide interfaces with directories. They may also automate the processing of echeck attachments such as invoices, remittance information and advice of payment documents, and they may provide mechanized interfaces to the user's accounting systems. The design of these systems will depend on the types of network services used, the degree of automation desired, the legacy accounting systems, and the design approach of the payer and payee system vendor. The
4.1. **Payer System**

Figure 8 shows a functional diagram of a payer system.

4.1.1. **Network Interface**

At the right side of Figure 8 is the network interface. This may be based on an email client, an email server or an SSL/HTTP client. Functions provided in the interface for a mail client system would include:

- firewall functions or other functions to protect the payee's operating system from penetration,
- POP3 (Post Office Protocol) mail client,
- S/MIME (Secure-Multipurpose Internet Mail Extensions) decryption of incoming mail and encryption of outgoing mail,
- mail logging or archiving.

4.1.2. **Incoming Invoice Processing**

Processing of incoming invoices is especially important for the case of commercial payments using echecks. A significant benefit of using echecks for payment is the ability to receive invoices and return them with the payment so that the payee knows how to accurately post the payments in accounts payable.
Information about the payee can be extracted and stored in the payee directory, and the invoice itself is passed to accounts payable until the payment is approved and due.

4.1.3. **Payee Directory**

The payee directory includes payee email addresses, payee encryption keys and certificates, and the payees' echeck certificates and account blocks if the payer wishes to write checks payable to the payee's account or key in order to uniquely identify the payee. These may have been received in an invoice or in a payment previously received from the payee.

4.1.4. **Accounts Payable Interface**

The design of the interface to accounts payable will depend on the payee's accounts payable system. These may vary from a paper interface, or comma delimited text files, to more sophisticated file transfers, database interfaces or APIs. When a payment is due it is initiated by information received from accounts payable.

4.1.5. **User Interface**

Functions provided via the user interface include:

- approval of payments initiated from accounts payable,
- manual initiation of check payments or corrections,
- entry of PINs to unlock the check signing and administration features of the electronic checkbook,
- review of the check register and initiation of any reports or audits based on its contents, and
- administration of the payee system.

4.1.6. **Check Writing**

Functions provided by check writing include:

- receiving information needed for the check such as payee, date, and amount,
- formatting the information into an FSML document ready to be signed by the electronic checkbook,
- appending the payee's certificates, account blocks, and any attachments or invoice blocks,
- unlocking the electronic checkbook using the PIN received from the user,
- interacting with the electronic checkbook to sign the echeck,
- updating the check register, and
- passing the signed echeck and attachments to the mail system along with payee directory information

4.1.7. **Check Register**

The check register contains a record of echecks written by the payee. It should be compared periodically with the secure log/register contained in the electronic checkbook.
4.2. Payee System

Figure 9 shows a functional diagram of a payer system. Many of the functions are the same as for the payer system, and in practice bank customers would have systems that include all functions needed to write, endorse and deposit echecks.

4.2.1. Check and Attachment Verification

After the incoming echeck and attachments have been received and optionally decrypted by the network interface the check and attachment verification function performs the following functions:

- receives an email and determines that it contains an echeck,
- parses the FSML,
- cryptographically verifies the signatures on the echeck, the account block, and certificates from a trusted root public key,
- evaluates the pay-to, date, account restrictions and other data contained in the echeck to determine that the echeck is properly formed, valid and intended for the payee,
- determines whether the echeck is a duplicate of one received in a previous transmission,
- displays any failures or errors to the user interface,
- detaches any attachments or invoices returned by the payer as an advice of payment, and
- makes the echeck available for endorsement and deposit.

It should be noted that echecks can be verified without use of the electronic checkbook hardware.
4.2.2. **Endorsement and Deposit Processing**

Endorsement and deposit processing does require the use of the electronic checkbook. The functions performed include:

- receiving information needed for the endorsement and deposit blocks,
- formatting the information into an FSML document ready to be signed by the electronic checkbook,
- appending the payee's certificates, account blocks, and any attachments or invoice blocks,
- unlocking the electronic checkbook using the PIN received from the user,
- interacting with the electronic checkbook to endorse each check,
- interacting with the electronic checkbook to sign the deposit,
- updating the check register, and
- passing the endorsed checks and deposit to the mail system for transmission to the payee's bank.

4.2.3. **Advice of Payment Processing**

Like the interface to accounts payable, the accounts receivable interface will depend upon the payee's accounts receivable system.

It should be noted that the payee can send any document of the payee's choosing, for example an XML document, to the payer as an attachment. When this document is returned to the payee intact, the payee can parse and decode the document to extract the information needed for posting the payment in their accounts receivable.

4.3. **Electronic Checkbook**

Figure 10 is a functional block diagram of the electronic checkbook that may have one of several forms:

- Consumers and small businesses will typically use smart cards.
- Type 2 PC Cards can be used in notebook computers that are already equipped with PCMCIA connectors. PC Cards can also offer higher speed interfaces, faster processors, larger secure check log/registers, and more sophisticated tamper resistance than smart cards.
- Cryptographic hardware can be attached to servers for high-volume applications. These units may also perform high-volume signature verification, as well as signing, in order to off-load processing from the servers.
- Software-only implementations can be provided to payees when the payee is enabled only to endorse and deposit echecks in his own bank. This reduces the startup costs until the payee wishes to be enabled for echeck writing as well, and it may be a suitable permanent solution for biller's accounting departments.

It provides for the internal generation of the signing key pair and for the storage of the private key and for their use by trusted cryptographic software routines contained in ROM. This prevents the
private keys from ever being used outside of the tamper resistant card, and prevents them from being extracted via the connector. The checkbook also provides for PIN access control and for card administrative functions, such as initialization. The checkbook will have a check log or register to provide a secure record of signatures made using the card.

Figure 10. Electronic Checkbook Card Functions

It is planned that consumer and small business check writing and endorsing will be done using smart card or PC Card electronic checkbooks. This satisfies four requirements:

1. The private signature key is held in a tamper-resistant, PIN-activated hardware, so that it is less accessible for theft and subsequent fraudulent check writing. This also minimizes the skill and diligence required of the least conscientious checking account holders. If the private signature keys were held in password-protected files on network-connected PCs, it is likely that many would be poorly protected by the account holders. Were hackers to acquire echeck signing keys and to begin perpetrating check fraud in significant amounts, the payees’ trust and willingness to accept echeck payments would be destroyed.

2. The card prefixes a random nonce and an automatically incremented check number to the electronic check. This guards against birthday attacks on the hashing and signature algorithm. It also ensures that every echeck will be unique, so that the payer’s bank can always dishonor second and later copies of a check, should such copies occur due to e-mail problems or deliberate copying.

3. The card logs certain data before hashing and signing the echeck within the card. This provides a secure log of echeck signing actions, which satisfies requirements for a receipt. It also provides a starting point for investigation if a Trojan Horse program has caused the card to sign a check different from the one that the payer thought was being signed.
4. The card contains personal information about the signer, such as name, address, telephone number, electronic mail address, etc., which was stored in the card at initialization, and which cannot be changed by the signer. However, under control of the signer, some or all of this information can be included in the electronic check, in order to satisfy the payee’s need for reliable identification of the payer.

Figure 10 shows three different interfaces for "echeck writing", "checkbook administration" and "checkbook initialization". These correspond to the electronic checkbook being enabled by one of three different PINs.

1. When enabled for echeck writing, the electronic checkbook can be used to sign echecks, endorse echecks, sign deposits, read out the echeck log/register, and change the echeck writing PIN.
2. When enabled for checkbook administration, the electronic checkbook can do all the functions of above, plus the check log/register items can be deleted, certificates and root keys can be added or deleted, the signer public key and signer personal data can be read out, and the administrative PIN can be changed.
3. When enabled for initialization at the bank all functions can be performed, including initializing the public key pair and the signer personal data.

The echeck writing and checkbook administration PINs are different so that the checkbook can be enabled for check writing without letting a Trojan Horse in the check writing application overwrite or delete the check log/register. Thus, for example, a point of sale terminal which writes a check for more than the amount showing on the terminal’s display, will be unable to erase the record of the inflated check from the checkbook’s log. User’s should only enter the signer administrative PIN and do log management functions when they can be reasonably sure that their application software and operating environment is trustworthy.

The two signer PINs should be forced to change upon first use to user-selected values. The bank initialization PIN is used to initialize the signer personal information, to unlock the checkbook when the user has forgotten his PINs, or when the checkbook has been locked due to PIN hacking countermeasures.

4.4. The Electronic Checkbook Interface

4.4.1. The Electronic Checkbook Application Programming Interface

The electronic checkbook will often be provided by the bank, just like paper check blanks, but others will often provide the echeck application and its integration into the users' accounting systems. Therefore, a standard API (Application Programming Interface) has been defined. It consists of a set of C-language function calls that are used by the echeck application to unlock the electronic checkbook and exercise all of its check writing and administration functions.

In a typical configuration for the Microsoft Windows operating system, the user will receive a smart card electronic checkbook, a smart card reader that attaches to a COM port, and a disk containing drivers and the electronic checkbook DLL.
Electronic checkbooks may vary in capability and sophistication. For example, verification of cryptographic signatures may be done on the card or in software in the DLL. Requirements for the electronic checkbooks identify levels of electronic checkbook implementations with various tradeoffs between hardware and software functionality, with a concomitant impact on cost.

From the perspective of the echeck application, all electronic checkbooks must appear the same and implement the same set of function calls.

### 4.4.2. The Electronic Checkbook Connector Interface

At this time, there is no definition of the electronic checkbook interface at the smart card connector.

When electronic checkbooks are used at point of sale terminals, then a connector level interface will need to be defined, including all protocol levels from electrical signals up to the application level.

A smart card may not be the best device for use at point of sale terminals. Alternatives may include electronic checkbook functionality built into PDAs (Personal Digital Assistants) such as the 3COM Palm III. In this case, infrared links may be used instead of electrical contacts. Use of a PDA for check writing may have security advantages over smart cards, since the user has an independent display and data entry capability, and the user does not have to trust the merchant terminal.

### 5. Bank Servers

Echeck servers in banks are used to receive echecks from customers by Internet email, process the echecks, and interface with legacy Demand Deposit Account systems that banks use to maintain checking account records. The design of the bank servers depends on the legacy systems, on implementation decisions regarding placement of functions in the new server or the legacy systems, and the design approach of the bank server vendor. The functional diagrams in the following sections are intended to explain new functions that must be carried out in the bank, but they do not dictate actual designs.

#### 5.1. Echeck Server at the Bank of First Deposit

Figure 11 is a diagram of functions that would typically be provided by an echeck server in the bank of first deposit. This server receives emails from payees that contain endorsed echecks and deposits. The echecks are processed and held in the echeck data base until they are extracted daily for clearing and settlement with other banks. Echecks that are "on us" and deposits are submitted to the DDA systems for processing. Problematic echecks and deposits are routed to the Research Workstation for manual analysis and intervention.

Beginning in the upper left, the mail containing echecks is received by SMTP mail processing after it has passed through the firewall and other security measures. If the mail is encrypted, it is decrypted using the bank's private decryption key. The sender's public key certificate is processed, and the secure mail directory is updated with any new certificate and public key information for the sender.
The FSML document is extracted from the received email, and it is logged. FSML documents that are duplicates caused by, for example, mailer resends can be suppressed at this point. Original FSML documents are parsed and decomposed into their individual blocks and data elements.

Cryptographic processing is done to verify signatures on the deposit, endorsements by the payee, and signatures by the payer. Signatures on the signer's X.509 certificates are verified, and the chain of trust is followed to a root public key known a priori to the server. Certificates and account blocks of the bank's own customers are also checked against the X.509 certificate and account data base in order to determine whether the certificate has been revoked or not and whether the account is still current.

Echecks are processed according to rules that apply to all echecks or that are determined by bank policies. Processing includes:

- Detect and discard echecks which are duplicates of previous echecks within a time interval equal to the duration until which echecks become stale (a shorter interval may be used for echecks drawn on other banks).
- Assign a bank stamp number to each item.
- Convert the echeck account numbers of deposits and on-us echecks to the actual paper checking account numbers (unless this account number mapping is done in the DDA system).
- Verify that the payee identification on the check is that of the endorser.
- Reject echecks deposited prior to the value date.
- Verify that the echeck does not violate restrictions in the account block or check block.
Deposits are proofed against the included checks and adjusted for checks which do not verify and must be returned.

Once processed, echecks are stored in the data base.

If there were errors in the echeck or deposit, or if the item meets specific bank rules (large amount, new account, etc), then the item is forwarded to the research workstation. The workstation provides displays to pinpoint why the item was rejected, including parsing problems, failure to verify signatures, failure to verify against processing rules, and other conditions.

Periodically a process is run to extract echecks and deposits from the data base for processing during the daily DDA systems run. Deposits and on-us echecks are sent to the bank's DDA system, and echecks drawn on other banks are sent for clearing and settlement. Current practice is to map the echecks into X9.46 files for exchange with other banks over non-Internet data networks.

The echeck server maintains a record of each echeck and deposit, so that it can correctly process returned echecks and deposits. Echecks are archived to permanent storage and kept for the same time as microfilms of paper checks. Echecks may be considerably compressed for archival storage, since there will be many instances of the same account blocks, certificates, etc. which may be stored once and referenced.

Bank servers must provide administrative reports regarding echecks and deposits processed and regarding the operation, integrity and security of the echeck server which are sufficient to meet bank auditing requirements.

Data can flow from the bank server, through the DDA systems and on to statementing so that additional information about the echeck can be included in the customer's statement. For example, payee and date written can be provided along with today's typical check number, amount, and date posted. Some customers receiving electronic statements may also wish to receive their "canceled echecks".
5.2. **Echeck Server at the Paying Bank**

Figure 12 shows the functions that would be performed at a bank receiving checks from a bank of first deposit.

The functions are similar to those in the bank of first deposit except that there is:

- no Internet interface
- no email processing, and
- no deposit processing.

However, this figure shows a second interface to clearing and settlement for the return of echecks returned for insufficient funds, no such account, and other causes.

5.3. **Direct Exchange with Due-To and Due-From Accounts**

Nearby banks that clear large numbers of paper checks often do so directly. Cash letter and bundle blocks have been defined in FSML for the purpose of direct presentment of echecks between bank servers.

6. **Checkbook Distribution and the Public Key Infrastructure**

Electronic checkbook distribution can vary considerably among banks so long as basic requirements are met. These include:
The steps in the process are as follows:

1. Marketing and sales contacts a customer who is interested in electronic checking.
2. The customer signs up for electronic checking, either added to an existing account or by opening a new account. The information collected from the customer includes the usual customer name, address, telephone, etc, as well as information about the PC or server hardware, operating system and email account that the customer will be using for electronic checking.
3. The information is entered in the bank administrative systems used to manage opening of new accounts.
4. Electronic checkbook issuance instructions are sent to the card initialization operation. Data includes all the information needed to initialize the signer personal data, the account block, and the X.509 certificate for the account. This requires that the bank administrative systems have assigned the electronic account number. Other information may be included to initialize use of the electronic checkbook card as a cash card, debit card or charge card.
5. The card initialization operation generates a key pair in the card, extracts the public key, and sends the account block and X.509 certificate information to the bank certification authority for signing.

6. The certification authority signs the X.509 certificate and the account block using the bank CA's private signing key, and returns the certificate and account blocks to card initialization.

7. The bank CA also sends the certificate and account block to the bank echeck server to update its data base with the pending account.

8. The certificates and account blocks are installed in the card, the card is printed, and the PINs are initialized. A PIN mailer is printed and sent to the customer. The card may be embossed, and the magnetic stripe may be written to enable other card functions.

9. The electronic checkbook, smart card reader, software and instructions are assembled in a kit and shipped separately to the customer.

10. Card initialization notifies the account administrative systems that the card has been sent.

11. The account administrative systems activate the account in the bank echeck server, possibly after receiving an email or other confirmation from the customer.

12. If the card is lost or stolen, the customer reports it to the bank. The bank CA revokes the certificate and notifies the bank server of the revocation. Other banks may be notified by CRL, and the bank may take other actions with respect to the account.

7. Conclusion

The electronic check is the first instance of a family of cryptographically signed financial documents. It proposes a flexible and extensible syntax, a flexible cryptographic signature mechanism, the use of tamper-resistant token cards to secure private keys and log transactions, and the use of an X.509 certificate public key infrastructure. It will reduce fraud and allow verification of signatures on every check. It is compatible with world wide web, Internet mail and many other transport systems, and these systems can be encrypted to provide privacy. It provides an excellent start towards shifting a wide variety of financial instruments and documents from paper to electronic form. This will reduce financial transaction overhead costs, enable new electronic commerce services, and allow banks to provide new and better services to their customer.